8-2015

Development of High Performance Molecular Dynamics with Application to Multimillion-Atom Biomass Simulations

Roland Schulz
University of Tennessee - Knoxville, rschulz3@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Roland Schulz entitled "Development of High Performance Molecular Dynamics with Application to Multimillion-Atom Biomass Simulations." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Biochemistry and Cellular and Molecular Biology.

Jeremy C. Smith, Major Professor

We have read this dissertation and recommend its acceptance:

Hong Guo, Xiaolin Cheng, Tongye Shen

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Development of High Performance Molecular Dynamics with Application to Multimillion-Atom Biomass Simulations

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Roland Schulz

August 2015

To family and friends!

Acknowledgements

I would like to thank my adviser Jeremy C. Smith and my committee Hong Guo, Xiaolin Cheng, and Tongye Shen.

All former and current members of the Center for Molecular Biology, in particular Benjamin Lindner, Loukas Petridis, and John Eblen.

All members of the GROMACS team, in particular Berk Hess, Erik Lindahl, Teemu Murtola, Szilard Páll, Christoph Junghans, and Mark Abraham.

The Genome Science and Technology program.

Funding from Intel Corporation and DOE, and computation resources provided by ORNL NCCS, NERSC and UTK NICS made this work possible.

And, of course, my family.

You could give Aristotle a tutorial. And you could thrill him to the core of his being. Such is the privilege of living after Newton, Darwin, Einstein, Planck, Watson, Crick and their colleagues. Richard Dawkins

Abstract

An understanding of the recalcitrance of plant biomass is important for efficient economic production of biofuel. Lignins are hydrophobic, branched polymers and form a residual barrier to effective hydrolysis of lignocellulosic biomass. Understanding lignin's structure, dynamics and its interaction and binding to cellulose will help with finding more efficient ways to reduce its contribution to the recalcitrance. Molecular dynamics (MD) using the GROMACS software is employed to study these properties in atomic detail. Studying complex, realistic models of pretreated plant cell walls, requires simulations significantly larger than was possible before. The most challenging part of such large simulations is the computation of the electrostatic interaction. As a solution, the reaction-field (RF) method has been shown to give accurate results for lignocellulose systems, as well as good computational efficiency on leadership class supercomputers. The particle-mesh Ewald method has been improved by implementing 2D decomposition and thread level parallelization for molecules not accurately modeled by RF. Other scaling limiting computational components, such as the load balancing and memory requirements, were identified and addressed to allow such large scale simulations for the first time. This work was done with the help of modern software engineering principles, including code-review, continuous integration, and integrated development environments. These methods were adapted to the special requirements for scientific codes. Multiple simulations of lignocellulose were performed. The simulation presented primarily, explains the temperature-dependent structure and dynamics of individual softwood

lignin polymers in aqueous solution. With decreasing temperature, the lignins are found to transition from mobile, extended to glassy, compact states. The low-temperature collapse is thermodynamically driven by the increase of the translational entropy and density fluctuations of water molecules removed from the hydration shell.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Molecular Dynamics

Molecular Dynamics (MD) is a simulation method which is widely used in biology and material science (Becker et al., 2001), since the first simulation in 1977 (McCammon et al., 1977). The Born-Oppenheimer approximation is applied and only the nucleus of the atoms are modeled explicitly. The forces between atoms are computed by multiple bonded and non-bonded force terms. Different force-fields might have different formulas for the force terms. All currently, widely used non-polarizable force-fields have formulas similar to:

$$\begin{aligned} V = \sum_{\text{bonds}} k_b(b-b_0)^2 + \sum_{\text{angles}} k_\theta(\theta-\theta_0)^2 + \sum_{\text{dihedrals}} k_\phi[1+\cos(n\phi-\phi_0)] + \\ \sum_{\text{improp}} k_\omega(\omega-\omega_0)^2 + \sum_{\text{UB}} k_u(u-u_0)^2 + \\ \sum_{\text{VDW}} \left(\frac{A}{r_{ij}^{12}} + \frac{B}{r_{ij}^{6}}\right) + \sum_{\text{charges}} \frac{q_i q_j}{\varepsilon r_{ij}} \end{aligned} \tag{1.1}$$

with the abbreviations improper dihedrals (improp), Urey-Bradley (UB), and van der Waals (VDW). The first five terms are only computed between bonded atoms and approximate the forces exerted on the nucleus by the electrons forming the chemical

bond. All k_x and x_0, where x is b, θ, ϕ, ω, and u respectively for each of the five terms, are constants which depend on the atom type. Each atom might have multiple atom types based on the chemical environment. A biological force-field contains the constants for all atom types required to simulate standard biological molecules such as proteins and DNA. The k_x constants determine the force strength and the x_0 constants the equilibrium position. The creators of the force-field determine the constants by a mixture of quantum-mechanical calculations and experimental results.

The last two terms represent the non-bonded forces. The van der Waals forces describe the dispersion forces and the forces originating from the Pauli exclusion principle. The forces between charges are the coulomb forces between partial charges. The partial charges are the net charges of the nuclei and the polarized electron clouds. If evaluated directly, the non-bonded forces are to be computed between all atom pairs. Such a direct evaluation has a computational complexity of $O(N^2)$. The $O(N^2)$ complexity makes it infeasible to simulate large systems. Out of the two components of the van der Waals force, the one which asymptotically approaches zero as a function of distance at a slower pace is the dispersion force ($\frac{1}{r^6}$). At around 1nm distance it is small enough that a cut-off at this distance is usually accurate enough. Only a switch or shift function is needed, as a modifier to the standard functional form, to guarantee a smooth first derivative, which is important for the integration. The Coulomb force decays significantly slower to zero as a function of distance; this is caused by the functional form $\frac{1}{r}$. As importantly, many commonly studied biological systems have charged atom groups such as phosphate groups for DNA or charged amino acids for protein. These groups of atoms, smaller than the size of the typical cut-off used for VDW, have a net charge. Without a net charge, only higher order terms (dipole, quadrupole, ...), with faster decaying functional form, would contribute to long distance coulomb forces. To accurately model the interaction between these charged groups of atoms, a method without cut-off is required.

The most commonly employed method to reduce the complexity to $O(N \log N)$ is Particle Mesh Ewald (PME, Darden et al. (1993); Essmann et al. (1995)). It employs a

splitting function to solve part of the Poisson's equation by a direct sum over particles and the remaining part in Fourier space. The direct space part is computed up to a cut-off distance of fixed size ($O(N)$) and the Fourier space part is computed utilizing the Fast Fourier Transform (FFT) method ($O(N \log N)$). An alternative method is the Reaction Field (RF) method. Its function form is derived from the assumption of a constant dielectricum with constant ε outside of a cut-off. For small, sparse, neutral molecules in water with $\varepsilon = \varepsilon_{\text{water}} \approx 80$ this is a very good approximation. For larger molecules it can provide very good agreement with PME.

1.2 Supercomputing

The current state of computing is dominated by parallelism. Moore's law, transistors double approximately every two years, is valid even after 50 years. Until about 2004 the extra transistors in each generation were used to increase single thread performance by increasing frequency of the CPU and increasing instruction level parallelism (ILP). This meant that the same code would run faster with each CPU generation; at most requiring the code to be recompiled. Because of power constraints, the maximum frequency and ILP has diminishing returns, the single thread performance has improved at a much lower rate since then, forcing CPU manufactures to place multiple cores on a single CPU to improve the total computing power with the available transistors. This requires software to make use of parallel hardware for good performance. Power efficiency is crucial for supercomputing because the cost for power and associated cooling is a large part of the total cost. Currently the most power efficient architectures are General-purpose computing on graphics processing units (GPGPU) and many-core CPUs. These are also those selected for the largest upcoming supercomputers. GPGPUs are based on graphics processing units (GPU) which were developed for 3D graphics. High-end GPUs are made by AMD and Nvidia. They typical have around 2000 cores and use single instruction, multiple thread (SIMT). With SIMT, 32 or 64 cores together execute an

instruction by operating on vector data. There are 60 group of threads, and each can execute an independent software thread. To date, the only high-end many-core CPU is the Xeon Phi by Intel. It has around 60 cores and each core can execute 16 wide vector instructions using single instruction, multiple data (SIMD). Compared to widely used desktop and server CPUs, the cores both in the Xeon Phi and in GPUs are significantly simpler. This allows more of them to be placed on one chip, saving power, but also implies that the serial performance is significantly lower than for standard CPUs. The more complex features in standard CPUs missing from GPUs and Xeon Phis are e.g. out of order execution, long execution pipelines, and complex branch prediction.

1.3 Programming Productivity

Scientific computing is immensely important for modern science. It requires software developed by scientists who have the domain knowledge of the science simulated, but also have knowledge about computer algorithms, programming languages, hardware, and parallel programming. At the same time, there is often very little formal training in development methodology and tools for computational scientist. In turn many best practices which have been applied for commercial software development over the last decade, have not been adapted by the majority of scientific software packages. Such best practices include test-driven development (TDD), integrated development environment (IDE), continuous integration, core review, and coding standards, and also older ones such as object-oriented programming.

1.4 Lignocellulosic Biomass

Understanding lignin's structure and dynamics is an important component of efficient biofuel production. The plant cell dry mass consists mainly of cellulose, lignin, and hemicellulose. Converting cellulose to biofuel is a well understood process and can

be accomplished on an industrial scale. Cellulose is first split by a cellulase into cellobiose which in turn is broken into glycose by another cellulase. The glycose is then fermented into ethanol. To obtain the cellulose from the biomass pretreatment is required. The lignin is the plant's natural defense against microbial digestion. Before cellulases can effectively split cellulase, lignin has to be separated from cellulose. This pretreatment is responsible for a significant fraction of the production cost of cellulosic ethanol from lignocellulosic biomass due to the associated energy requirements as well as capital and operating costs.

Chapter 2

GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers

This chapter is revised based on a paper with the same title as the chapter submitted to SofwareX authored by Abraham, M. J; Murtola T.; Schulz R.; Páll, S.; Smith J.C.; Hess B.; and Lindahl E.. My primary contributions to this paper include (i) most of the work for Xeon Phi (ii) parts of the C++ transition (iii) adding the Random123 library (iv) adding code review and continuous integration to the development process, and (v) adding the VMD plug-in for file IO.

Abstract

GROMACS is one of the most widely used open source and free software codes in chemistry. As of version 5, it achieves better performance than ever before through parallelization on every level from SIMD registers inside cores, multithreading, heterogeneous CPU-GPU acceleration, and state-of-the-art parallelization with 3D

domain decomposition. Several new algorithms provide highly efficient simulations on any hardware available, highly compressed data storage and a rich set of analysis tools. The code provides advanced techniques for free energy calculations, and supports ensemble-level parallelization both through built-in replica exchange and the separate Copernicus framework.

2.1 Motivation and Significance

Molecular dynamics (MD) has conquered chemistry and several other fields by providing spatial and temporal resolution not available in experiments. Simulations have become more accurate with better force fields, they easily sample molecular motions on the μs scale, and ensemble techniques make it possible to study millisecond scale processes such as protein folding. This is achieved by evaluating the interactions of millions of particles for billions of time steps, which can require extraordinary amounts of computational hardware and time - the scientific quality of the result is often proportional to the amount of sampling. The huge application potential has led to implementations of MD in many software packages, including GROMACS (Pronk et al., 2013), AMBER (Case et al., 2005), NAMD (Phillips et al., 2005), CHARMM (Brooks et al., 1983), LAMMPS (Plimpton, 1995), and Desmond (Bowers et al., 2006). The commoditization of advanced computational techniques by these packages is an important reason for the wide adoption of MD today.

Many implementations (including ours) provide high performance when using large numbers of processors on supercomputers, but a key focus for the development of GROMACS is the fundamental assumption from economic science that *resources are scarce*: No matter how many cores are available, minimizing resource usage makes it possible to run more simulations, e.g. through ensemble methods. GROMACS aims to address this by providing the highest possible absolute performance, but also by combining it with the highest possible efficiency on any hardware to make best use of scarce resources. The package runs fast on every single architecture present in

the Top500 supercomputer list, as well as on embedded systems and everyday laptop computers.

In contrast to many other computational challenges, applications in MD typically have an intrinsically fixed problem size. When studying a protein system with 30,000 atoms it is not relevant that a virus comprising 10 million atoms would scale better. Therefore, weak scaling performance is typically not of primary concern. However, it is critically important to reduce the amount of computer time per unit simulation through optimization, or by improving strong scaling. This improves "time to solution," and also the efficiency, measured as the science performed per amount of hardware or power consumed. While some applications need long individual trajectories, there are also many scientific questions that can be answered by using several trajectories, and for these the overall efficiency will be higher by executing independent simulations in parallel.

However, strong scaling of MD is a very difficult software engineering challenge that requires synchronization of computation, coordination, and communication phases down to 100 μs for hundreds of thousands of cores.

Development of GROMACS has long been characterized by dedication to high absolute simulation throughput on commodity hardware; we want to improve the strong scaling so that both the *maximum achievable* and *real world* throughput increases. Hardware advances have been breathtaking, but re-engineering the software to use the new capabilities has been very challenging, and even forces us to reconsider some of the most fundamental MD concepts including the neighbor lists used to track spatial interactions. This paper will briefly recount some historical properties of GROMACS, and then report on recent improvements in GROMACS 4.6 and 5.

2.2 Software Description

GROMACS has grown into a very large software project with almost two million lines of code. For a detailed description of the historical development and many algorithms

included in the engine, we refer the interested reader to the previous papers published (Páll et al., 2015; Pronk et al., 2013; Bekker et al., 1993b; Berendsen et al., 1995a; Lindahl et al., 2001; van der Spoel et al., 2005a; Hess et al., 2008). For developers, one of the most important changes is that GROMACS 5 is the first release that has moved to C++. While many parts of the code remain in C and it will take a few years to complete the transition, this has led to improvements in code modularity, handing of memory and errors, and enabled much better Doxygen developer documentation and unit testing.

2.2.1 Software Architecture

The efficient parallelization in GROMACS is based on an "eighth shell" spatial domain decomposition (Bowers et al., 2005, 2007) to partition the simulation system over multiple hardware units in a way that preserves locality of reference within each domain. Using this data parallelization, a single program multiple data (SPMD) implementation maps each domain to an MPI rank, each of which can in practice have access to various kinds of hardware. Internally, all systems are described with triclinic unit cells, which makes complex geometries such as rhombic dodecahedron, truncated octahedron or hexagonal boxes supported in all parts of the code. This can improve performance up to 40% compared to the same water thickness around a solute in a rectangular box (Fig. 2.1). Dynamic load balancing between domains is performed in all three dimensions in triclinic geometry; this is critical for high performance. Fig. 2.1 shows how the larger computational load due to torsions and angles in the protein compared to water leads to significant differences in domain size in the upper left part.

Long-range electrostatics is handled by the particle-mesh Ewald (PME) method (Darden et al., 1993) by using dedicated MPI ranks for the lattice summation and a two-dimensional pencil decomposition (Pronk et al., 2013) for the required 3D-FFT.

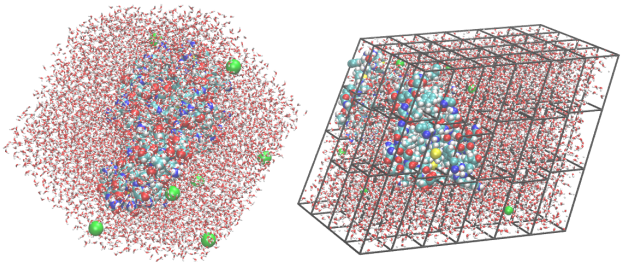


Figure 2.1: *Left:* The protein lysozyme (24,119 atoms) in a compact unit cell representation corresponding to a rhombic dodecahedron with close-to-spherical shape. *Right:* Internally, this is represented as a triclinic cell and a load-balanced staggered 6x4x3 domain decomposition grid. The PME lattice sum is calculated on a uniform grid of 6x4 MPI ranks (not shown).

Historically, GROMACS has made use of MPI for domain-level parallel decomposition across nodes, and later CPU cores too, and supplied hand-tuned assembly kernels to access SIMD (single instruction, multiple data) units where available. However, the run-time overheads of the former and the development-time cost of the latter were not sustainable, and there was also the need to incorporate accelerators (such as GPUs) into the parallelization strategy. GROMACS 4.6 introduced a native heterogeneous parallelization setup using both CPUs and GPUs. There are two important reasons for still including the CPU: First, the advanced domain decomposition and load balancing would be very difficult to implement efficiently on GPUs (which would hurt scaling). Second, we see it as a huge advantage that *all* algorithms are available in all simulations, even the esoteric or new ones not yet ported to GPUs, and the heterogeneous acceleration makes it possible to completely hide the hardware from the user. There is only a single GROMACS binary, and it will use all available hardware as efficiently as possible.

To make this possible, a new algorithm for evaluating short-ranged non-bonded interaction was implemented, based on Verlet lists with automatic buffering (Páll and Hess, 2013). This recasts the traditional Verlet algorithm to suit modern computer hardware, which permits highly efficient offload of short-ranged work on SIMT-based (simultaneous multithreading) GPUs, as well as efficient SIMD-based CPU

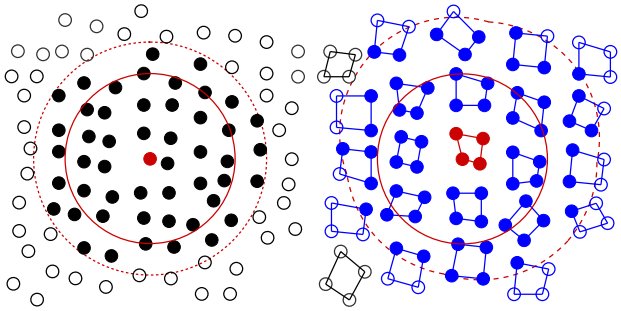


Figure 2.2: *Left:* A classical Verlet implementation treats all j-particles within an interaction radius (red) of the central red i-particle, and adds a buffer, also called "skin" (dashed red). Particles outside the buffer (unfilled) are omitted. *Right:* The $M \times N$ scheme builds lists of clusters of N particles (blue), where at least one particle in each cluster is within the buffered interaction radius of any particle in the central red cluster. This envelope has an irregular shape (dashed red), and has an implicit additional buffer (unfilled blue circles) from those particles in clusters where only some particles are within the nominal buffer range. Actual interactions are based on particle distances (red circle, only one shown).

implementations. This works well, since the essential qualities of data locality and reuse are similar on both kinds of hardware. This is an important architectural advance, since the same code base and algorithms can be used for all hardware. The key innovation was to transform the standard formulation of the Verlet algorithm that uses lists of particle-particle pairs into lists of interacting small clusters of nearby particles, and to choose the sizes of those clusters at compile time to match the characteristics of the SIMT or SIMD target hardware. This means there is no requirement for the compiler to recognize the opportunity for vectorization; it is intrinsic to the algorithm and its implementation. Additionally, the cluster sizes are easily adjustable parameters allowing to target new hardware with relatively low effort. Fig. 2.2 shows how the Verlet scheme re-casts the idea of a particle-based pair list into a list of interacting clusters. Fig. 2.3 illustrates the flow of data in kernels executing on processors of different SIMD widths or GPUs.

Unlike the old "group" scheme, there is no need for special kernels optimized for common molecules such as water. The searching can schedule kernels that will evaluate van der Waals interactions only on the first half of atoms in a given cluster;

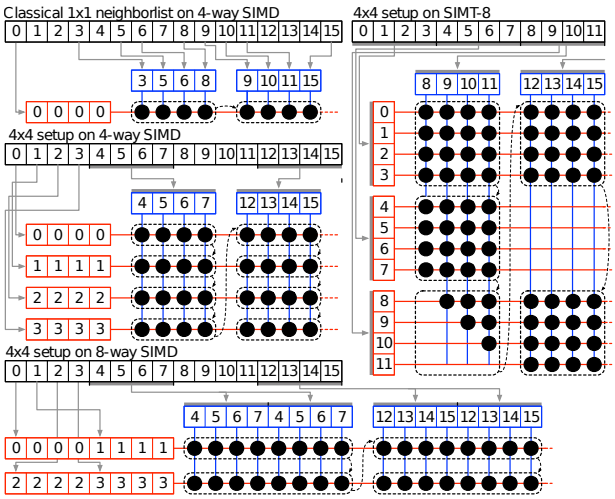


Figure 2.3: Illustration of the classical 1×1 neighborlist and the 4×4 non-bonded cluster algorithm setups on processors with different SIMD widths. All numbers are particle indices in a pair list, black dots are interaction calculations, and gray arrows indicate loads. The content of SIMD registers for i- and j-particles (cf. Fig. 2.2) are shown in red and blue, respectively. Dashed black lines show the computation units, with dotted black arrows indicating their order of execution. The 4×4 setup calculates 4 times as many interactions per memory operation. Unlike the 1×1 setup, the 4×4 setup does not require data shuffling in registers.

this runs faster on domains where only some atoms have such interactions, which includes most water models in current use. Naturally, if whole clusters do not have van der Waals or Coulomb interactions, their interactions are evaluated by kernels that skip the corresponding computation entirely. Branches are unavoidable in short-ranged MD kernels, as the model physics permits only interactions within a certain distance to contribute. Implementing such code is most efficient when using selection or branch instructions that produce null results when the interaction should not contribute. This is also useful for other kinds of interaction exclusions used in MD.

The maturation of SIMD intrinsics in compilers made this possible to implement in a new higher-level fashion that retains the performance of the previous raw hand-tuned assembly. To achieve this, we have implemented a SIMD abstraction module that permits us to develop CPU non-bonded kernels in a way that is nearly agnostic about the SIMD width and feature set of the hardware upon which they will run. In particular, we have designed an extensive new internal SIMD math library in both single and double precision that completely avoids both table lookups and integer instructions (which are not available for all SIMD instruction sets). This means that porting to new CPU architectures is a straightforward job of implementing the interface of the SIMD module using the intrinsics suitable for the new CPU, and the old non-bonded kernels can use them correctly. Further, several other modules in GROMACS now use the same SIMD layer and derive the same benefits for performance portability. Crucially, this has reduced the total size of the nonbonded kernels to only a few hundred lines of C, while simultaneously supporting many more SIMD instruction sets.

In particular for SIMD and GPU acceleration, GROMACS makes extensive use of strength-reduction algorithms to enable single precision, including a single-sum implementation of the virial calculation (Bekker et al., 1993a). Some molecular simulation packages always compute in double precision; this is available in GROMACS for the few kinds of simulations that require it, but, by default, a *mixed precision* mode is used, in which a few, critical, reductions are performed in double precision.

Other high-performance implementations (Case et al., 2005) use mixed precision to a larger extent.

The offload model for acceleration creates the need for large groups of atoms in the same spatial region to be treated in the same neighbor search. This conflicts with the former GROMACS model of mapping each CPU core to an MPI rank, and thus a separate domain of atoms close in space. Typically, a CPU has many more cores than it has accelerators, the inefficiency of scheduling separate work for each domain on the accelerator was high, and the existing limitations on the minimum sizes of domains was also problematic. To alleviate this, OpenMP-based multi-threading support was added to GROMACS to permit multiple CPU cores to work on a single domain of atoms. This allows for domains to be larger and thus the overall efficiency to be greatly improved. The resulting OpenMP parallelism is also useful for running on CPU-only hardware, thus extending the strong-scaling limit with hybrid MPI/OpenMP.

The PME algorithm commonly used for molecular simulations is able to shift workload between the short- and long-ranged components with moderate efficiency, while preserving the quality of the model physics. This permits GROMACS 5 to automatically balance the workload for optimal performance. This is particularly useful for the offload model implemented in GROMACS 5 because best throughput is typically obtained when few resources lie idle. Further, when using multiple nodes for a single simulation, the long-ranged component of the PME calculation needs to do global communication for the 3D FFT. This partly drives our choice of which work to offload; doing PME work on a current generation accelerator in a simulation across multiple nodes would accrue latency from data transfers both across the network and from host to device and this would eliminate any performance gain.

One major weakness of the current accelerator-offload implementation in GROMACS is that accelerators are idle once the forces are computed and transferred back to the CPU. Typical schemes for integrating the forces to update the positions often need to enforce holonomic constraints on degrees of freedom such as bond lengths, and such implementations normally feature either iteration or inter-rank communication, which

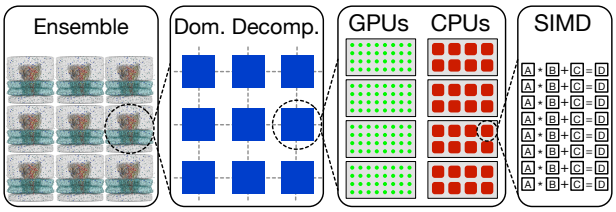


Figure 2.4: Multi-level parallelism in GROMACS. SIMD registers are used to parallelize cluster interaction kernels or bonded interactions in each core, and OpenMP multithreading is used for parallelism inside spatial domains while nonbonded interactions are handled by GPUs or other accelerators. MPI and load balancing is used to decompose a single simulation into domains over many nodes in a cluster, and ensemble approaches are used to parallelize with loosely coupled simulations. High performance requires that software targets each level explicitly.

does not suit an offload model of accelerator usage. Overcoming such limitations is a key target for future improvements.

The end result in GROMACS 5 is an elaborate multi-level parallelism (Fig. 2.4) that maps individual particle-particle interactions to SIMD or SIMT lanes and offloads short-ranged work during force calculations to GPU accelerators. Long-range and bonded interactions are evaluated on the CPU, which uses OpenMP to split work over multiple cores for a single spatial domain, and MPI to communicate between these domains. This design is able to make effective use of all of the available resources for typical PME simulations on typical hardware. However, the design works less well if the hardware is too unbalanced; GROMACS 5 performance will typically be optimal with comparable expenditure on CPU and accelerator, and as many CPU sockets as accelerators.

2.3 Software Functionalities

GROMACS is free software distributed under LGPLv2.1, which even allows linking into commercial applications. It requires only standards-conforming C99 and C++98 compilers, and CMake version 2.8.8. Various external libraries are either bundled for convenience, or can be detected (e.g. MKL) or even downloaded automatically

(e.g. FFTW) by CMake. Portability is assured via extensive automated continuous integration testing using Jenkins, deployed on Windows, MacOS and Linux, using multiple versions of compilers including those from Intel, GNU, Microsoft, LLVM and IBM. GROMACS supports NVIDIA GPU architectures with compute capabilities 2.0 and later, and the new SIMD module provides native support for a total of 13 different architectures including all x86 flavors (from SSE2 through Xeon Phi, AVX2 and the still unreleased AVX-512F/ER), PowerPC A2 (BlueGene/Q), Sparc VIIIfx (K computer), ARM Neon, IBM VMX (Power7) and VSX (Power8). The latest version can even run inside a browser supporting Google Native Client.

Every single commit during GROMACS development is subject to mandatory code review and automatic regression tests, unit tests and static code analysis before it is added to the public git repository. While the released code is tested on an even larger set of architectures, this makes even the rapidly moving development branch uniquely stable.

2.3.1 Simulation Capabilities

Simulations with leap-frog Verlet, velocity Verlet, Brownian and stochastic dynamics are supported, as well as calculations that do energy minimization, normal-mode analysis and simulated annealing. Several techniques are available for regulating temperature and/or pressure. Both SHAKE (Ryckaert et al., 1977) and P-LINCS (Hess et al., 1997; Hess, 2008) are available for enforcing holonomic constraints, and the latter can be combined with virtual interaction sites (Berendsen and van Gunsteren, 1984) to eliminate enough fast degrees of freedom to allow 5 fs time steps. All widely used molecular mechanics force fields are supported, including a total of 15 flavours of AMBER, CHARMM, GROMOS and OPLS. Several community-supported force fields are also available. Non-standard functional forms are supported through user tables.

Simulations may employ several kinds of geometric restraints, use explicit or implicit solvent, and can be atomistic or coarse-grained. `mdrun` can run multiple simulations as part of the same executable, which permits generalized ensemble methods (Mitsutake et al., 2001) such as replica-exchange (Hansmann and Okamoto, 1997; Sugita and Okamoto, 1999). Non-equilibrium methods, such as pulling and umbrella sampling, are available, as well as highly powerful alchemical free-energy transformations, and essential dynamics (Amadei et al., 1993). Many popular simulation file formats can be read natively, or via a VMD plug-in (Humphrey et al., 1996).

The code scales *down* to a few tens of atoms per core (when only using CPUs), but there will always be practical limits on the degree of parallelism achievable. A typical 150,000-atom system has about thirty million particle-particle interactions per MD step, which will not scale to a million-core system because communication and book-keeping costs will dominate. Thus, as core counts continue to grow, we expect ensemble-level parallelism to play an increasingly important role in MD algorithm development. The Copernicus framework has been developed alongside GROMACS to serve this need and scale to tens of thousands of simulations (Pronk et al., 2014). It currently supports free-energy calculations, Markov state modeling, and the string method using swarms (Pan and Roux, 2008) (`http://copernicus.gromacs.org`).

2.3.2 A Parallel Analysis Framework

A new C++ framework for developing GROMACS analysis tools has been introduced, which makes it easy to write new tools that require only a simple loop over all trajectory frames. The framework also provides reusable components for grid-based neighbor searching and common data processing tasks like histograms. Some tools for computing basic geometric properties (distances and angles), as well as surface area calculation, have been converted to the new framework, though much work remains to

achieve the full benefits of the new scheme. Future development also aims to support analysing single trajectory frames in parallel, and Python bindings.

2.3.3 New Simulation Features

Just as PME has eliminated cutoff artefacts for electrostatics, there has been increasing attention to cutoff problems and van der Waals interactions. While dispersion corrections alleviate some issues, the fundamental problem is that complex systems such as membranes are neither homogeneous nor isotropic. GROMACS 5 includes a new, very accurate, Lennard-Jones PME implementation (Wennberg et al., 2013) whose implementation is only 10-20% more expensive than short cutoffs in GROMACS, and to the best of our knowledge about an order of magnitude faster than any other alternative. It works for both geometric and Lorentz-Berthelot combination rules, and should enable much more accurate membrane simulations, free energies, and improved force-field parameterization.

Other new features include Andersen-style thermostats, the Adaptive Resolution Sampling scheme (Fritsch et al., 2012) for multi-scale models, Hamiltonian replica exchange, simulated tempering and expanded-ensemble methods (Lyubartsev et al., 1992), rotation of groups with the non-equilibrium pulling module (Kutzner et al., 2011a), a new computational electrophysiology module Kutzner et al. (2011b) that can swap molecules from one side of a membrane to the other, and support for the Interactive Molecular Dynamics Stone et al. (2001) protocol to view and manipulate ongoing simulations. The high-quality "counter-based" parallel random number generator Random123 Salmon et al. (2011) is now used. New bonded interactions were introduced for coarse-grained simulations (Bulacu et al., 2005). Flat-bottomed position restraints were added to avoid perturbing models unecessarily.

GROMACS 5 also comes with a new highly flexible and efficient compressed file format - TNG (Lundborg et al., 2014). This improves on the previous best-in-class XTC trajectory compression by further exploiting domain knowledge and multi-frame

compression, it adds features such as containers for general simulation data, digital signatures, and provides a library to which tool developers may link.

2.4 Impact

In addition to the thousands of publications using GROMACS every year, one of the most exciting parts of free software is how other people put it to use in ways not anticipated. GROMACS has long been deployed in the Folding@Home distributed computing project (Shirts and Pande, 2000), and it is frequently used for metadynamics together with PLUMED (Tribello et al., 2014). Coarse-grained force fields such as MARTINI Marrink et al. (2007) use the GROMACS infrastructure to implement mesoscale physics models that access otherwise impossible scales of time and distance. Databases of topology file inputs and associated thermochemical results have begun to appear Caleman et al. (2012); van der Spoel et al. (2012), and several online services can produce coordinates, parameters and topologies for GROMACS simulations (Zoete et al., 2011; Malde et al., 2011). Extending and reusing parts or all of GROMACS is explicitly encouraged.

2.5 Performance & Scaling

GROMACS can scale to the largest machines in the world when using gigantic systems, and detailed benchmarks are available on the website. For this work, we want to illustrate the efficiency with more challenging heterogeneous benchmarks used in recent studies: First a very small voltage sensor (VSD) embedded in a united-atom lipid bilayer in a hexagonal box (45,700 atoms) Henrion et al. (2012), and second a complete ion channel (GluCl) embedded in a larger united-atom bilayer (142,000 atoms) (Yoluk et al., 2013). All simulations use PME and initial cut-offs of 1.0nm. All bonds were constrained with the LINCS Hess (2008) algorithm, and the VSD uses virtual interaction sites constructed every step to extend time time step to 5

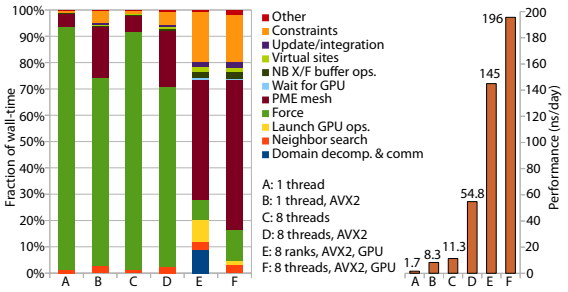


Figure 2.5: The anatomy of GROMACS performance. *Left:* The reference setup (A) spends the majority of wall-time in short-range force evaluations, but as SIMD (B), OpenMP (C, D), and GPU (E, F) acceleration are enabled this drops tremendously even on a workstation. With GPUs (E, F), the CPU computes a relatively small amount of bonded interactions. Relying on multi-threading (F) when using GPUs is more efficient than SPMD parallelization with domain-decomposition (F) even though multi-threading in cache intensive code like PME is challenging. Using GPUs *Right:* The absolute performance is orders-of-magnitude higher with accelerations, which explains the larger fraction of CPU time spent on constraints, update, and reciprocal space PME.

fs. A stochastic velocity-rescaling thermostat was used (Bussi et al., 2007). Fig. 2.5 shows how the fraction of CPU cycles spent on force evaluation in the VSD system drops dramatically when adding SIMD and GPUs. Case "E" reflects the problem with multiple MPI ranks on the CPU and effectively time-sharing the GPU, which introduces decomposition overhead. With GPUs, there is only a small component left for bonded forces; the CPU primarily evaluates the PME mesh part. The absolute performance is much higher with acceleration (explaining the larger fractions for constraints and PME), as illustrated in the second panel. These results were obtained on a single-socket desktop with an 8-core Core-i7 5960X and a single NVIDIA GTX980 GPU. With both SIMD, GPU and OpenMP acceleration, the desktop achieves close to 200ns/day for the VSD. Fig. 2.6 shows absolute performance for the larger GluCl system on CPU-only and GPU-equipped Cray clusters. Despite the much faster CPUs with AVX2 support on the XC40, the older XC30 nodes paired with K20x GPUs beat it handily. With similar CPUs and K40 GPUs, we expect accelerated clusters to deliver about 3X the performance of CPU-only ones.

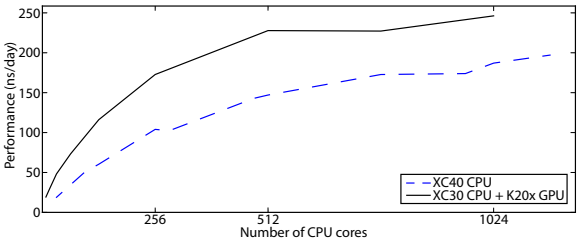


Figure 2.6: Absolute scaling performance on a GluCl ion channel of 142,000 atoms, 2 fs time steps without virtual sites, with PME and initial cutoffs 1.0 nm, running on Beskow (32 Haswell CPU cores per node, Cray XC40) and Piz Daint (8 Sandy Bridge CPU cores + Tesla K20X per node, Cray XC30).

2.6 Conclusions

The recent efforts to maximize single-core and single-node performance show benefits at all levels, which makes it even more impressive that the relative scaling has also improved substantially. The enhancements described above boost user throughput on all kinds and quantities of hardware, but they were focused on mature technologies. There are many other approaches open for future performance improvements to GROMACS. In particular, an implementation that expresses the algorithm in fine-grained tasks that can be preempted when high-priority work is available, and automatically balanced between otherwise idle executors seems very attractive. Task parallelism has been quite successful e.g. in NAMD with the Charm++ programming model (Phillips et al., 2005), but with the latency constraints we are targeting we need a lower-level implementation. Currently the most promising one is Intel's Threading Building Blocks (Pheatt, 2008), because it is also C++98 and open source. With TBB, we expect to be able to retire our home-grown thread-MPI implementation and avoid requiring OpenMP support, as well as opening the door to even higher absolute performance and scaling. To get started, download the code from `http://www.gromacs.org`.

Chapter 3

Scaling of Multimillion-Atom Biological Molecular Dynamics Simulation on a Petascale Supercomputer

This chapter is revised based on a paper with the same title as the chapter published in J. Chem. Theory Comput., 2009, 5, 2798-2808 authored by Schulz, R.; Lindner, B.; Petridis, L. ; and Smith, J. C. My primary contributions to this paper include (i) developing of the approach for the simulation including the selection of the Reaction Field method, (ii) most of the method validation, (iii) all improvements to GROMACS, (iv) most of the simulation setup excluding the psfgen modifications, and (v) collecting the scaling data.

Abstract

A strategy is described for fast all-atom molecular dynamics simulation of multimillion-atom biological systems on massively-parallel supercomputers. The strategy is developed using benchmark systems of particular interest to bioenergy research,

comprising models of cellulose and lignocellulosic biomass in aqueous solution. The approach involves using the Reaction Field (RF) method for the computation of long-range electrostatic interactions, which permits efficient scaling on many thousands of cores. The use of the RF produces molecular dipole moments, Kirkwood G factors, other structural properties and mean-square fluctuations for the benchmark systems in excellent agreement with those obtained with the commonly-used Particle Mesh Ewald method. With RF, 3M- and 5M-atom biological systems scale well up to ∼30k cores, producing ∼30ns/day. Atomistic simulations of these very large systems for timescales approaching the microsecond would therefore appear to now be within reach.

3.1 Introduction

Molecular Dynamics (MD) simulation is a powerful tool for the computational investigation of biological systems (Becker et al., 2001). Since the first MD study of a protein in 1977, which simulated < 1000 atoms for < 10ps (McCammon et al., 1977), significant progress has been made in the time and length scales accessible and it is now common to probe systems of $\sim 10^5$ atoms on timescales of ∼100ns. This increase in scope has allowed many processes of biological interest to be characterized. However, there is clear interest in further extending both the time- and length-scales beyond those currently accessible.

Recent algorithmic (Phillips et al., 2005; Hess et al., 2008; Plimpton, 1995; Bowers et al., 2006) and hardware developments have allowed MD studies to be extended to multimillion atom systems (see, for example, Refs. Freddolino et al. (2006); Sanbonmatsu and Tung (2007)). Current supercomputers, such as the "Jaguar" Cray XT5 at Oak Ridge National Laboratory used for the present study, are beginning to assemble over 10^5 cores and in this way reach petaflop nominal speeds. However, the challenge for MD, and other applications, is to achieve efficient scaling up to

$\sim 10^4 - 10^5$ cores *i.e.*, the simulations are limited by the parallel efficiency of the MD algorithms, that is their ability to run in parallel on many thousands of processors.

The most computationally demanding part of MD simulation of biological systems is the treatment of long range interactions, which in non-polarizable force fields is represented by Coulomb and van der Waals (Lennard-Jones) terms (for a review see Ref. Mackerell (2004)). While the van der Waals interaction is almost always truncated at a cutoff distance R_{vdw}, the electrostatic interaction extends to longer ranges. A common method to treat the electrostatics is to directly calculate the Coulomb interaction for any pair of atoms separated by less than another cut-off distance R_{coul}, and outside this distance to calculate the interactions with the Particle Mesh Ewald (PME, Darden et al. (1993); Essmann et al. (1995)) method (assuming periodic boundary conditions are applied to the system). By using an Ewald summation to split the Coulomb interaction into a short-range part that converges quickly in real space and a long-range part that converges quickly in reciprocal space the PME method reduces the computational cost of N particles interacting with each other from $O(N^2)$ to $O(N \ln N)$. The reciprocal space sum is performed by using the fast Fourier transformation (FFT).

Full electrostatic treatment via the PME method presents a performance barrier on massively parallel computers, arising from the state-of-the-art implementation of PME, which requires two FFT steps. The FFT algorithm in turn requires one or two global transposes which, on a message passing system, is inherently limited by the bandwidth and latency of the network. As more nodes are used to simulate a system, each MD time-step can be calculated faster and thus the time between communications becomes shorter. If the time for the global transpose is of the same order of magnitude as the computation time, the required communication becomes a bottleneck for the parallel efficiency. The same reasoning explains why, when running on the same number of cores, the parallel efficiency of a large system (*e.g.* 1 million atoms) is much better than that of a small system (*e.g.* 1 thousand atoms): the time needed to compute a single time step on a single processor is much longer in case of a

large system. Therefore, for a large system, many more cores can be used before the communication bottleneck occurs. As a result, larger systems can often be simulated at about the same speed (in ns/day) as smaller systems.

An alternative method to PME, that avoids the electrostatics bottleneck, is the Reaction Field (RF, Gunsteren et al. (1978); Tironi et al. (1995)). In RF it is assumed that any given atom is surrounded by a sphere of radius, R_{rf}, again within which the electrostatic interactions are calculated explicitly. Outside the sphere the system is treated as a dielectric continuum. The occurrence of any net dipole within the sphere induces a polarization in the dielectric continuum, which in turn interacts with the atoms inside the sphere. The RF model allows the replacement of the infinite Coulomb sum by a finite sum modified by the reaction field, and therefore limits the parallel scaling less than the PME method.

The present paper outlines a strategy for fast and accurate all-atom simulation of multimillion atom biomolecular systems. The benchmark systems used in the present study are cellulose in water and models of lignocellulosic "biomass". Lignocellulosic biomass is a complex material composed of crystalline cellulose microfibrils laminated with hemicellulose, pectin, and lignin polymers (Cosgrove, 2005).In recent years there has been a revived interest in biomass structure, as biomass offers a potentially abundant and cheap source of sugar for industrial biofuel production (Himmel et al., 2007). Due to its complexity, lignocellulose poses significant challenges to MD simulation. Among these are the characteristic length-scales (Å-µm) and time-scales (ns-µsand beyond) of events pertinent to the recalcitrance of biomass to hydrolysis into sugars (Himmel et al., 2007). To access these length and time-scales standard MD protocols must be modified to scale up to massively parallel machines.

Two technical problems are addressed. Firstly, we compare the accuracy of MD using PME and RF on the benchmark systems, and, secondly we examine scaling of MD of large systems on a petascale supercomputer. In previous comparative studies, simulations using RF were found to yield similar structural properties as simulations using PME for solvated biomolecules such as RNA and proteins (Walser

et al., 2001; Nina and Simonson, 2002; Gargallo et al., 2003). However, to our knowledge a detailed comparison between RF and PME with respect to dynamical properties, which are likely to be affected by electrostatics, has not been performed. Furthermore, in simulations on bulk water using RF dipole correlations were found to be incorrect (Mathias et al., 2003; van der Spoel and van Maaren, 2006).

The present comparative studies show that the examined properties derived using PME are well reproduced using the computationally less demanding method of RF. Scaling benchmarks on multimillion systems show that the use of the RF drastically improves the parallel efficiency of the algorithm relative to PME, yielding ∼30ns/day. Consequently, microsecond time-scale MD of multimillion atom biomolecular systems appears now within reach.

3.2 Methods

3.2.1 Simulation Setup

The simulations were performed using cellulose (Kuttel et al., 2002) and lignin (Petridis and Smith, 2009) force fields parameterized for the CHARMM energy function (Mackerell et al., 1998) using GROMACS 4.0.4 (Hess et al., 2008) as the MD software.

The GROMACS simulations were performed with the electrostatic treatments RF, PME-cutoff, PME-switch, Shift and Switch (see Table 3.1). The analytical expression for electrostatic potential, V_{rf} with the RF method is:

$$V_{rf} = \left(1 + \frac{(\epsilon - 1)\, r^3}{(2\,\epsilon + 1)\, {r_c}^3}\right) r^{-1} - 3\,\frac{\epsilon}{r_c\,(2\,\epsilon + 1)}, \tag{3.1}$$

where ϵ is the dielectric constant outside the radius r_c and r is the distance separating two charges. In Switch and Shift a function S is added to the Coulomb force F_c, giving a total force $F_t = F_c + S$. S is a third-degree polynomial acting over interatomic distances r where $R_1 < r < R_{coul}$ and is zero otherwise, R_{coul} being the cut-off

Table 3.1: Sets of benchmark simulations performed. Each set comprises five 20ns-trajectories initiated from the same structure, but with a different initial velocity distribution.

Simulation Index	Electrostatic Treatment
1	PME with switch
2	PME with cut-off
3	RF
4	Shift

radius (van der Spoel et al., 2005b). R_1 is zero for Shift and corresponds to the switch-on distance for the Switch method (in this study $R_1 = 0.8$nm and $R_{coul} = 1.2$nm, more details on the simulation parameters follow in the next paragraphs). The polynomial is constructed so that $S(r_1) = S'(r_1) = F_t(r_c) = F_t'(r_c) = 0$. The Switch, Shift, RF and Coulomb functions are shown in Figure 3.1. The Switch electrostatics was immediately found to produce severe artefacts, including a strong suppression of the fluctuations of the heavy atoms of the cellulose. Consequently the switch simulation was not further considered for detailed analysis. We suspect the switch-induced errors to have been enhanced by the periodicity in the fibril.

The I_β allomorph of cellulose was simulated with the initial atomic coordinates taken from Ref. Nishiyama et al. (2002). This cellulose structure has two chains per triclinic unit cell which will be referred to as the "origin" and the "center" chains. The shape of the fiber was chosen as proposed in Ref. Ding and Himmel (2006). Figure 3.2 shows structural details of the model. Details on the models of the lignin molecules are presented elsewhere (Petridis et al., 2009).

For the simulations in which the effects of varying the electrostatic model were examined, the cellulose contained 80 monomers per chain and 36 chains and was solvated with 70656 TIP3P (Jorgensen et al., 1983) water molecules, producing a model totaling 272556 atoms. A triclinic box was used with a 60°angle between the two short box vectors. The systems were equilibrated for 1ns and then simulated for 20ns with a time step of 2fs. For each simulation set-up 5 simulations with

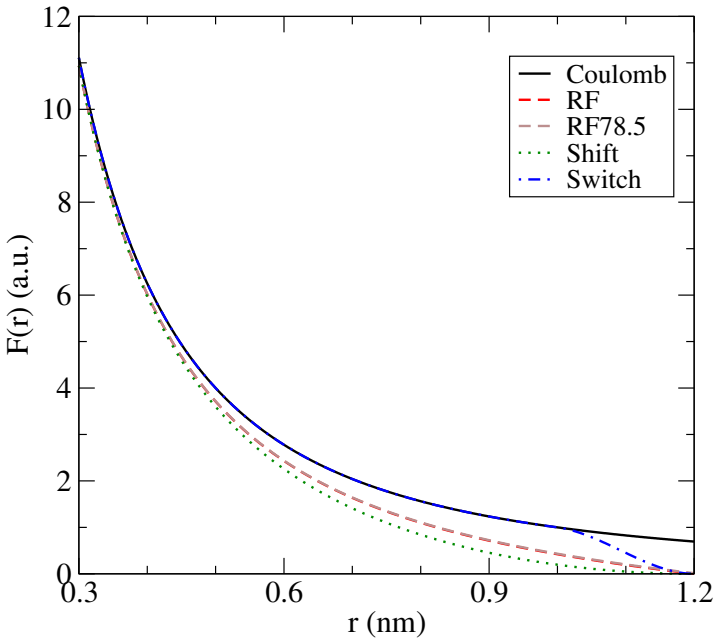


Figure 3.1: Coulomb force as a function of the distance between opposite charges. "Coulomb" is the Coulombic force without modification. "RF" is the reaction field with $\varepsilon = \infty$ outside the cut-off radius. For "RF78.5", $\varepsilon = 78.5$ outside the cut-off radius. "Shift" and "Switch" are computed as described in Ref. van der Spoel et al. (2005b). The switch distance after which the Coulomb function is altered is 1nm.

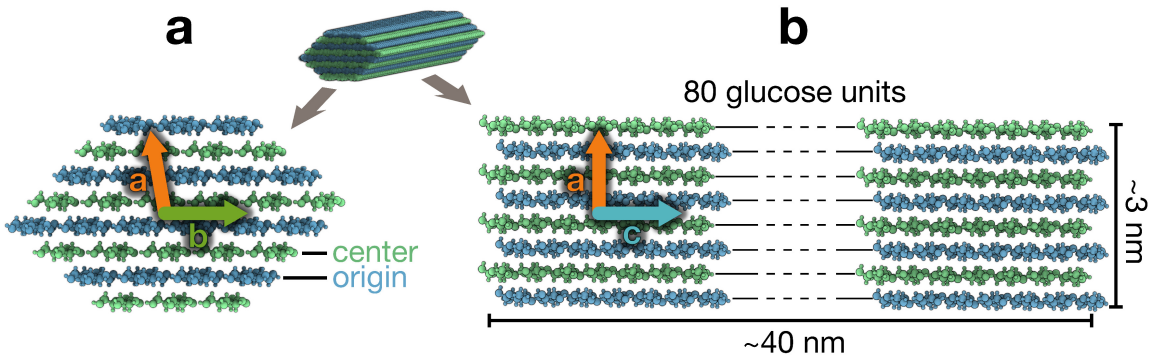


Figure 3.2: The model of the simulated cellulose fibril showing (a) the cross-section and (b) a side perspective. The fibril consists of 18 origin chains (blue) and 18 center chains (green). The axes of the unit cell are also indicated.

different initial velocities were run. Neighbor-searching was performed every 10 steps. All bonds were constrained using LINCS (Order: 3, Iterations: 2; Hess (2008)). Temperature coupling was performed with the Nosé-Hoover (Hoover, 1985) algorithm (τ = 1ps) during equilibration and with the Berendsen (Berendsen et al., 1984) algorithm (τ = 0.1ps) during production. Pressure coupling was performed with Berendsen algorithm (semi-isotropic, τ = 4ps) during equilibration and with Parrinello-Rahman (isotropic, τ = 4ps; Parrinello and Rahman (1981)) during production.

For the RF simulation, $\varepsilon_{rf} = \infty$ was used with the so-called Reaction-Field-Zero method which uses spline interpolated tables instead of the analytical expression (van der Spoel et al., 2005b). For the "RF", "Shift" and "PME with Switch" runs a neighbor-list-search distance of 1.5nm, a electrostatic and VDW cut-off of 1.2nm and a switch distance of 0.8nm were used. For "PME without Switch" a neighbor-list-search distance of 1.2nm, a electrostatic cut-off of 1.2nm, a VDW cut-off of 1.0nm, and a VDW switch distance of 0.8nm were used.

In a first analysis step, the simulations were inspected visually. This inspection showed that a strong artefact can arise in the case where only a small buffer region is employed between the cut-off radius and the neighbor list search distance. To

determine the optimal width of the buffer region, a series of simulations was performed varying the width from 0 to 0.3nm in 0.1nm steps. Simulations using non-PME electrostatics and with buffer regions < 0.3nm exhibited artificial deterministic linear translation of whole cellulose fibers along their axes with a speed of ~ 1m/s. Thus, for all further analysis a buffer region of 0.3nm was used for non-PME electrostatics.

For the supercomputing performance comparisons a system was constructed of lignocellulosic biomass containing 52 lignin molecules each with 61 monomers, the same cellulose fibril as described above and 1,037,585 TIP3P water molecules, totaling 3,316,463 atoms. All simulation settings, apart from bond constraints, were the same as the RF settings for cellulose given above. All bonds involving hydrogens were constrained with LINCS (Order: 4, Iterations: 1). For the sole purpose of extending the scaling tests to a larger system, an additional model system containing 64,000 dipeptide molecules (GLY-PRO) and 1,280,000 water molecules, totaling 5,376,000 atoms was also constructed. The system was simulated with the same protocol and parameters as the 3.3M atom lignocellulose system. The detailed system setup is described in Ref. McLain et al. (2008). For the PME simulations in Figure 3.9 the NAMD multiple time step method was used, in which the long-range electrostatics is computed only every third step and, in addition, a smaller buffer was used.

Topologies were generated in CHARMM (Brooks et al., 1983) and converted using a locally-modified version of psfgen (Gullingsrud et al. (2006), see appendix B for details). The correctness of the converted topology and force field was checked by comparison with CHARMM and NAMD (see appendix B). All analysis was performed using the tools provided by GROMACS (Lindahl et al., 2001; van der Spoel et al., 2005a). The NAMD trajectories were converted for analysis to GROMACS format and reordered with catdcd (Gullingsrud, 2009). Molecular drawings were made with QuteMol (Tarini et al., 2006).

3.2.2 Supercomputer Performance Measurements

The performance tests were run on JaguarPF, a Cray XT5 massively parallel processing (MPP) computer with over 150,000 Opteron 2.3GHz cores. JaguarPF has a LINPACK performance of over one Petaflop and a SeaStar 2+ interconnect. The internal timings of GROMACS 4.0.4 and NAMD (CVS version) were used. The IO time was included in the benchmarks. For RF with GROMACS all parameters were as described in the system setup Section 3.2.1. For PME with NAMD a neighbor-list search distance of 1.35nm, a multiple timestep method with full electrostatic frequency of 3 and steps-per-cycle of 24 was used. Because the IO time was found to be impacted by latency problems caused by Lustre scaling (due possibly to the Meta data server), to reduce the impact of the Lustre performance, only the three fastest of 12 runs were used in the performance measurements. Each MD run was limited to a wall clock time of 10min. For the thermostat and barostat the total energy and virial were computed every 10 steps. The calculation of the total energy/virial requires an MPI_Allreduce communication and therefore more frequent updates would limit the scaling. For the domain decomposition (DD) the 12,288 cores were arranged in a 3D 96x16x8 grid. The load balancing works by changing the volume of the DD cells relative to each other. For the minimum DD cell length, 0.77, 0.68, 0.34 of the average length were used for X,Y,Z respectively.

3.3 Results

3.3.1 Comparison of Simulations with Different Electrostatic Methods

As will be discussed in Sect. 3.3.2, fast MD simulation of the 3.3-million-atom lignocellulose system can be obtained using the Reaction Field method for treating the electrostatic interactions. This section is devoted to examining the accuracy of RF for the biomass test systems. For this, structural and dynamical properties are

compared in simulations using different electrostatic methods. The particular choice of properties for comparison is based on their structural importance and anticipated sensitivity to possible electrostatic artefacts.

In order to investigate the dependence of dynamical properties on the chosen electrostatics method, the set of MD simulations listed in Table 3.1 was analyzed. The system of a cellulose fibril in aqueous solution, *i.e.* without lignin, was chosen for this comparison. Lignin was omitted since significantly longer trajectories would be required for the convergence of dynamical properties, due to its amorphous character, thus complicating the comparison.

Quantities were calculated that are expected to be particularly sensitive to electrostatics. Two functions probing electrostatic-induced structure and dynamics are the total dipole moment of the fiber and the Kirkwood function between dipoles of different chains, the latter providing information on the distance-dependent correlation of molecular dipoles. Finally, three specific dihedral angles were selected for comparison due to their structural importance in cellulose.

Dipole Correlation

In the first comparison, shown in Figure 3.3, the Kirkwood factor, $G_k(r)$, of cellulose is presented. $G_k(r)$ is given by (Oster and Kirkwood, 1943):

$$G_k(r) = \sum_{r_{ij}<r} \frac{\vec{\mu}_i \cdot \vec{\mu}_j}{|\vec{\mu}|^2}\,, \tag{3.2}$$

where $\vec{\mu}_i$ and $\vec{\mu}_j$ are the electric dipole moments of glucose chains i and j, respectively, and r_{ij} is the distance between their centers of mass. $G_k(r)$ is a measure for the orientational ordering of the dipole moments of the cellulose chains in the fibril. It is clear from Figure 3.3 that the RF method is in very good agreement with the PME method, contrasting with Shift, for which $G_k(r)$ is much lower. The spread of the Shift profiles arises from differences between the individual simulations in the set.

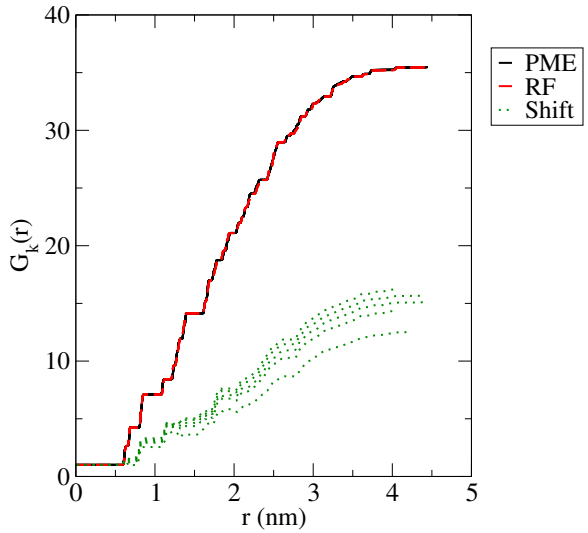


Figure 3.3: Distance-dependent Kirkwood factor (see Eq. 3.2). The two sets performed with PME are shown as indistinguishable black/solid lines. The RF set is red/dashed lines and the Shift set is green/dotted lines. The profiles of simulations with PME and RF are almost identical, implying very good agreement between the two methods.

In earlier work significant discrepancies were found between PME and RF78.5 (*i.e.* RF with $\epsilon_r = 78.5$) in the simulation of bulk water (Mathias et al., 2003; van der Spoel and van Maaren, 2006). We therefore investigated the cause of these discrepancies. To reproduce the earlier results, a simulation setup of bulk water as described in Ref. van der Spoel and van Maaren (2006) was constructed. The resulting Kirkwood factor for water is shown in Figure 3.4 for four distinct electrostatic treatments with this setup: (a) PME, (b) RF with an infinite dielectric constant ϵ, termed RF in Figure 3.4, (c) RF with $\epsilon = 78.5$ and (d) RF with $\epsilon = 78.5$ and an atom-based cut-off. For RF (b) interpolation of tabulated values was used and for RF78.5 (c and d) the analytical expression of RF was used directly.

The RF method with $\epsilon = \infty$ shows the best overall agreement with the PME. For RF78.5 the Kirkwood function is very different from PME with a deep minimum around the cut-off distance, agreeing with the previous observations (Mathias et al., 2003; van der Spoel and van Maaren, 2006). We performed several additional simulations (not shown) to find the reason for this difference of $G_k(r)$ for RF78.5. It turns out that the discrepancy arises from the combination of the neighbor-list search with the behavior of the analytical RF expression (Eq. 3.1). It is possible to simulate with an atom-based neighbor list, by splitting the water into three charge groups. Using this atom-based cut-off and updating the neighbor list at each step, the simulation of RF78.5 is found to agree well with RF and PME.

GROMACS calculates the electrostatic interaction for all atom-pairs included in the neighbor-list. The distance between atoms for which the electrostatic interaction is calculated can be larger than the cut-off distance in two cases: for a group-based cut-off, or for neighbor-list search frequencies ¡ 1/step. In the former case, only the group center needs to be within the cut-off distance for all atoms of the group to be included in the list. In the latter, it is sufficient for the atom to be within the distance at the time of the neighbor-list search, even if it moves outside afterwards. The analytical equation (Eq. 3.1) of RF78.5 is negative for distances longer than the cut-off distance. Using a spline interpolated table for RF78.5 instead of the analytical

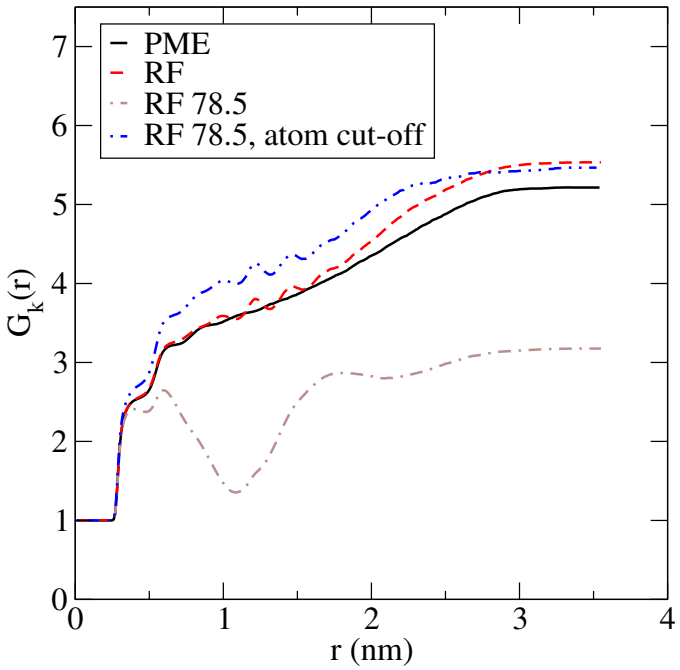


Figure 3.4: Distance-dependent Kirkwood factor (Eq. 3.2) for PME and RF with $\varepsilon = \infty$ and $\varepsilon = 78.5$, where ε is the dielectric constant outside the cut-off radius. RF 78.5 uses the group cut-off produces artefacts corrected by using an atom based cut-off and updating the neighbor-list at each step.

expression, as Reaction-Field-Zero does by default, allows the interaction for these distances to be set to zero. Simulation with this table is found to give very simular results compared to MD with an atom-based neighbor-list with search frequency of 1/step. Consequently, we conclude that the earlier observed difference between RF and PME (Mathias et al., 2003; van der Spoel and van Maaren, 2006) arises from the negative interaction of atom-pairs separated by distances longer than the cut-off distance, caused by the neighbor-list search.

Total Dipole Moment

A further useful test for global changes in dipolar correlation is the total dipole moment for a given macromolecule. Therefore, this should serve as a further benchmark for the accuracy of the electrostatic methods. As seen in Figure 3.5 the conclusions drawn from this comparison agree with those from Kirkwood G factor: RF and PME show similar features (RF yielding a total dipole moment about 1% lower than PME), whereas the Shift method results in a 50% lower magnitude and has slower convergence.

RMSF and RMSD. General dynamical properties examined include the Root-Mean-Squared Fluctuations (RMSF) and the modes resulting from Principal Component Analysis (PCA). The RMSF is a measure of the fluctuations of atoms around their equilibrium structure and PCA provides information on the major collective modes of motion. Both properties are commonly calculated in biomolecular simulations and were therefore checked for reproducibility.

Figure 3.6 shows the difference between the time-averaged RMSF of each atom in the cellulose fibril computed with the RF method minus the RMSF computed with the PME method ($\Delta(RMSF)_{RF}$). Also shown is the RMSF difference between the Shift and PME methods ($\Delta(RMSF)_{Shift}$). The overall good agreement between RF and PME is observed once more: RF enhances fluctuations slightly (with a more pronounced effect for the atomic indices in the range $35,000$ to $40,000$), but Shift

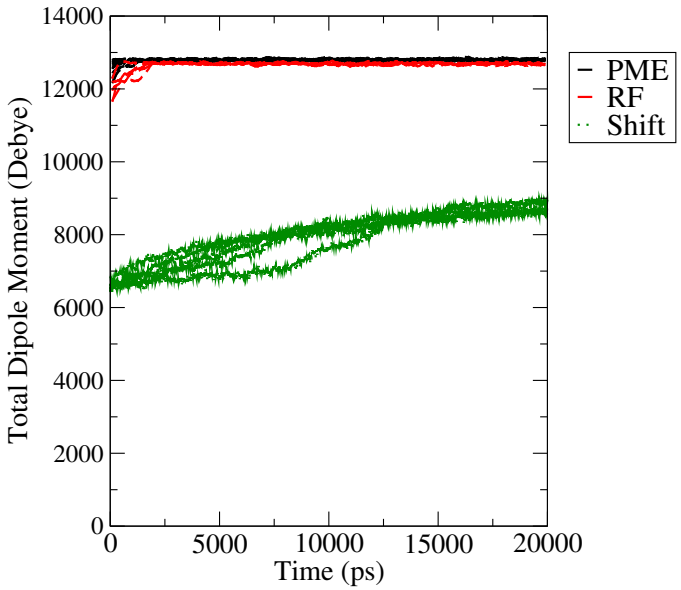


Figure 3.5: Total dipole moment of the cellulose fibril plotted. The two sets performed with PME are indistinguishable black/solid lines, the RF set in red/dashed lines and the Shift set in green/dotted lines. The profiles of simulations with PME and RF are almost identical.

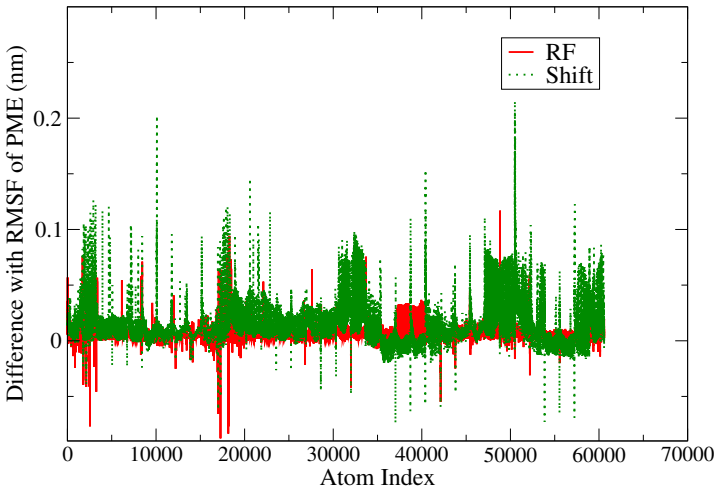


Figure 3.6: $\Delta(RMSF)_{RF}$ and $\Delta(RMSF)_{Shift}$ as defined in Section 3.3.1 for all atoms in the cellulose fibril (atomic index on x-axis).

leads to a much stronger deviation from the PME behavior. In contrast, the RMS displacement as a function of time, shown in Figure B.1 in the appendix B, shows little difference between the electrostatic treatments. Also in the appendix B it is shown that the amplitudes of the eigenvalues obtained from PCA of the trajectories using the three methods are similar.

Dihedral Angles.

The final test focuses on important local structural features of crystalline cellulose. Two sets of dihedrals are examined, as indicated in Figure 3.7. The particular relation of these dihedrals with respect to cellulose structure is discussed in detail in Refs. Matthews et al. (2006); French and Johnson (2004). The ω dihedral (O6-C6-C5-C4) determines the configuration of the primary alcohol group (Matthews et al., 2006), which affects the hydrogen bonding between adjacent glucose chains within a cellulose fiber and therefore is a main determinant for the crystalline phase. When the alcohol lies on the plane of the five-membered glucose ring ($\omega = -60°$ or $\omega = 180°$), single monomers preferentially hydrogen-bond to partners within the (010) crystal plane, whereas, when the primary alcohol points perpendicular to the five-membered

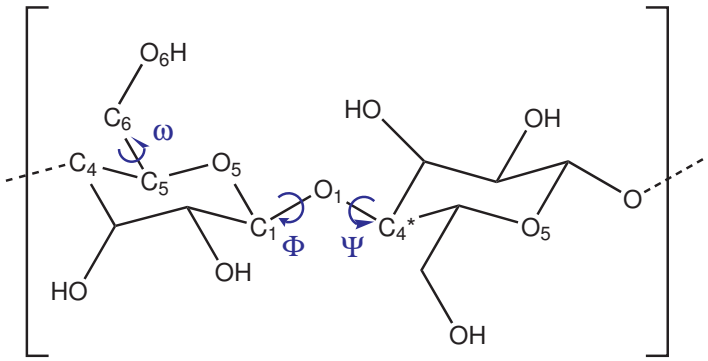


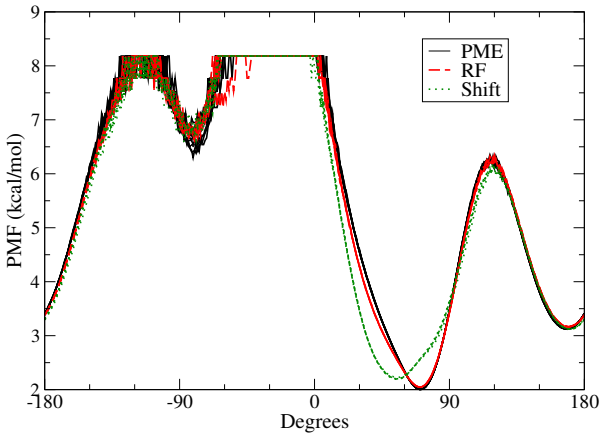
Figure 3.7: Sketch of cellobiose, the repeating unit of cellulose, indicating the three important dihedrals: the primary alcohol ω dihedral and the Ψ- and Φ- dihedrals.

ring plane ($\omega = 60°$) inter-sheet hydrogen bonds are formed. The $\Phi-$ and Ψ- angles (O5-C1-O1-C4* / C1-O1-C4*-O5*, where * marks atoms on the succeeding monomer) describe the twisting between two consecutive monomers and probe for the frustration in twisting behavior of isolated glucose chains induced by the fiber structure. Unlike the previous properties, these dihedral measures were not necessarily expected to be especially sensitive to differences in electrostatic treatment. They do, however, play an important role in the structure of cellulose. It is therefore of interest to determine weather their Potentials of Mean Force (PMF) are not significantly affected by variation of the electrostatic treatment.

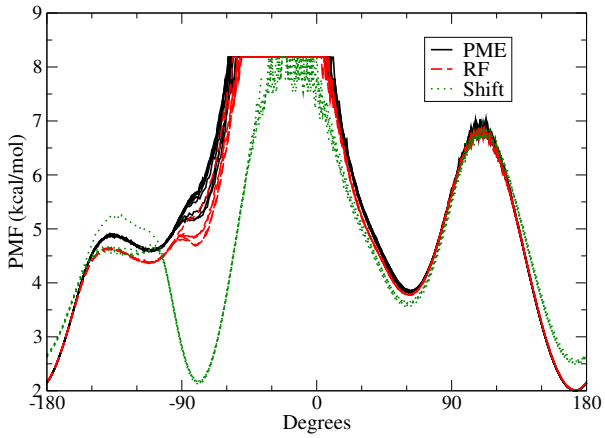
The PMFs were calculated according to the equation:

$$W(\theta) \;=\; -k_B T \log P(\theta)\,, \quad \theta = \{\omega, \Phi, \Psi\} \tag{3.3}$$

where θ is the dihedral angle in question and $P(\theta)$ is the associated probability distribution. Since the I_β crystal phase of cellulose has two distinct chains per unit cell, a total of six PMF calculations was performed: for each of the three dihedrals (ω, Φ and Ψ) the PMF was calculated for the center and origin chains. The resulting plots are shown in 3.8.



(a) Origin Chain



(b) Center Chain

Figure 3.8: Potentials of mean force for the primary alcohol dihedral ω =O6-C6-C5-C4: (a) results from all 36 origin chains and (b) results from all 36 center chains.

The PMFs for the primary alcohol dihedrals follow the same trend as the previous benchmarks *i.e.*, there is good agreement between the RF and PME methods, but not with the Shift method. We note that comparison of the profiles is only meaningful at the relatively low-energy regions that are adequately sampled. In the PMF for the origin chains in Figure 3.8a, the RF and PME profiles are almost indistinguishable. However, with the Shift method the global minimum moves from 70° to 50°. The difference between Shift and PME is even more pronounced in the PMF for the center chains (Figure 3.8b), for which Shift introduces a new minimum at −80°, which is only a weak shoulder in the PME calculations.

It is of interest that in the crystal structure of cellulose all primary alcohols have $\omega = -60°$ (Nishiyama et al., 2002). The transition from $\omega = -60°$ to $\omega = 180°$ observed during the MD simulation is as expected and has been reported in previous MD studies Matthews et al. (2006). The origin of the transition is that the force field employed (Kuttel et al., 2002) was parameterized for glucose in water and favors the $\omega = 180°$ conformation. Curiously, the Shift method appears to "correct" this short-coming of the force field and the $\omega = -60°$ conformation is populated in the center chains. However, this effect is probably a cancellation of errors. The present test concerns not the accuracy of the force field with respect to experiment, but rather a comparison between methods for treating long-range electrostatics.

The PMF for the Φ and Ψ dihedrals shows little variation between the simulations using the three electrostatic treatments and is given in Figure B.3 and B.4 of the appendix B.

In summary, the RF electrostatics method is found to accurately reproduce simulation results performed with the widely-used PME method. From the set of tests performed it can be therefore inferred that no artefacts are introduced by the proposed RF methodology.

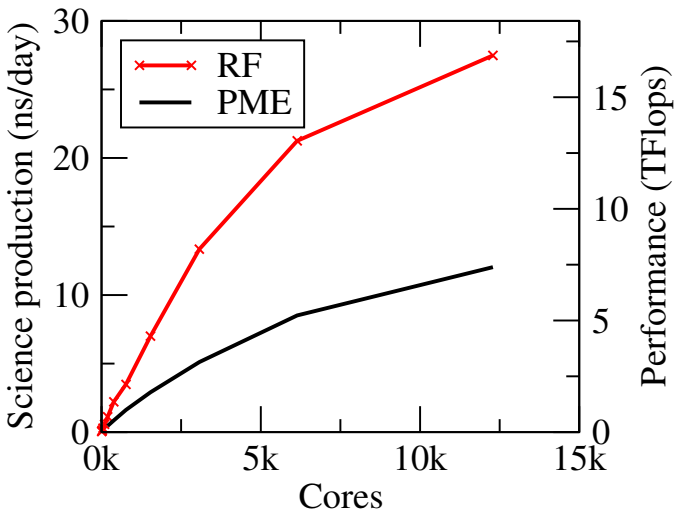


Figure 3.9: Strong scaling of 3.3 million atom biomass system on Jaguar Cray XT5 with RF using GROMACS. With 12,288 cores the simulation produces 27.5ns/day and runs at 16.9TFlops. As a comparison the performance of PME with NAMD is shown.

3.3.2 Scaling

The parallel efficiency of the RF MD simulation is now evaluated by considering strong scaling. In weak scaling the ratio of system size (*i.e.*, here, number of atoms in the system) to the number of cores used in the simulation is held constant, whereas in the strong scaling, the system size (number of atoms) in the system is held constant, while the number of cores used varies. The strong scaling of the 3.3 million atom MD simulation of lignocellulose using the RF on the ORNL "Jaguar" Cray XT5 is shown in Figure 3.9. For this system GROMACS scales well to 12,288 cores and achieves 27.5ns/day, running at 16.9TFlops. This performance is made possible by the good scaling of the RF and a fast particle-particle Streaming SIMD (Single Instruction, Multiple Data) Extensions (SSE) compute kernel running for the lignocellulose system at 4GFlops per Opteron core.

The RF also improves the parallel efficiency of the MD simulation of even larger systems. Figure 3.10 shows the strong scaling of a 5.4-million atom peptide solution

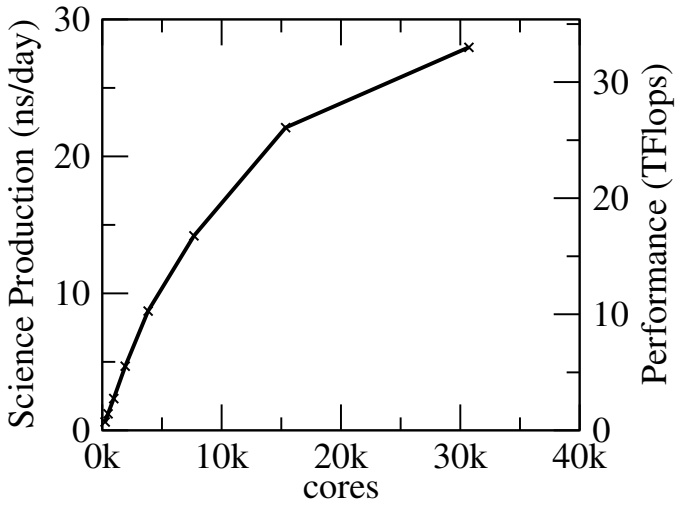


Figure 3.10: Strong scaling of 5.4 million atom system on Jaguar Cray XT5. With 30720 cores 28ns/day and 33TFlops are achieved.

test system. The same production of 28ns/day is obtained, this time scaling well to 30k cores.

The load balancing is most critical for the scaling to several thousand cores, as the load per volume of each domain-decomposition cell is not equal. The primary cause of load imbalance is the difference in computational complexity (in required Flops) between the solvent and the solute. The dynamical load balancing in GROMACS works by changing the domain-decomposition cell size as a function of the load imbalance. Each cell dimension is independently changed, causing the cells to move relative to each other, which leads to a staggered configuration. Optimization of the dynamical load balancing for cases with highly-staggered configurations allowed us to improve the average load imbalance from 200% to 75%, leading to a 44% improvement of the performance. This improvement resulted in the code obtaining the same production in ns/day using half the cores that were used prior to the improvement.

To highlight the computational benefit of using the RF, the scaling of a simulation of the 3.3 million lignocellulose system using the PME method is also shown in Figure 3.9. The PME simulation was run using NAMD, since this MD application is known to have good parallel efficiency (Schulten et al., 2008). To ensure a "fair"

comparison between the two electrostatics methods, some of the parameters of the PME simulation were adjusted to improve its performance (see Section 3.2.1 in the Methods for details). We stress that the aim of this benchmark is a comparison between electrostatic treatments and not between different MD applications. Two different applications were used simply because a direct comparison of simulations using different electrostatics method using one application is presently not possible: NAMD does not currently have RF implemented and GROMACS does not yet have an efficiently-scaling PME implemented (*i.e* using 2D decomposition).

The significant difference in the parallel efficiency of the PME and RF electrostatics methods, demonstrated in Figure 3.9, can be understood by examining the weak scaling of the parallel FFT required for PME, shown in Figure 3.11. The FFT is a new, improved implementation, the technical details of which are presented in appendix B.1. The Inset of Figure 3.11 shows that the new FFT is faster than the FFTs from LAMMPS-FFT (Plimpton, 2004), FFTE 4.0 (Takahashi, 2004) and FFTW 3.2 (Frigo and Johnson, 2005). In ideal weak scaling the time, t_f, required to perform one FFT step, indicated by the height of the bars in Figure 3.11, would remain constant as the number of cores used in the calculation increases from 16 to 38400. In practice, however, Figure 3.11 shows that parallel FFT calculations show poor weak scaling, with t_f increasing dramatically on a large number of cores. This increase is a result of the large increase of the required communication time (MPI-1+MPI-2 in Figure 3.11). Since, in weak scaling, the number of cores is proportional to the size of the simulated system, Figure 3.11 demonstrates that the PME method becomes computationally inefficient for large systems.

3.4 Discussion

This paper presents a strategy for efficient scaling of biological systems on massively parallel supercomputers. The key element of the strategy is to compute the long-range electrostatic interactions with the reaction field (RF) method.

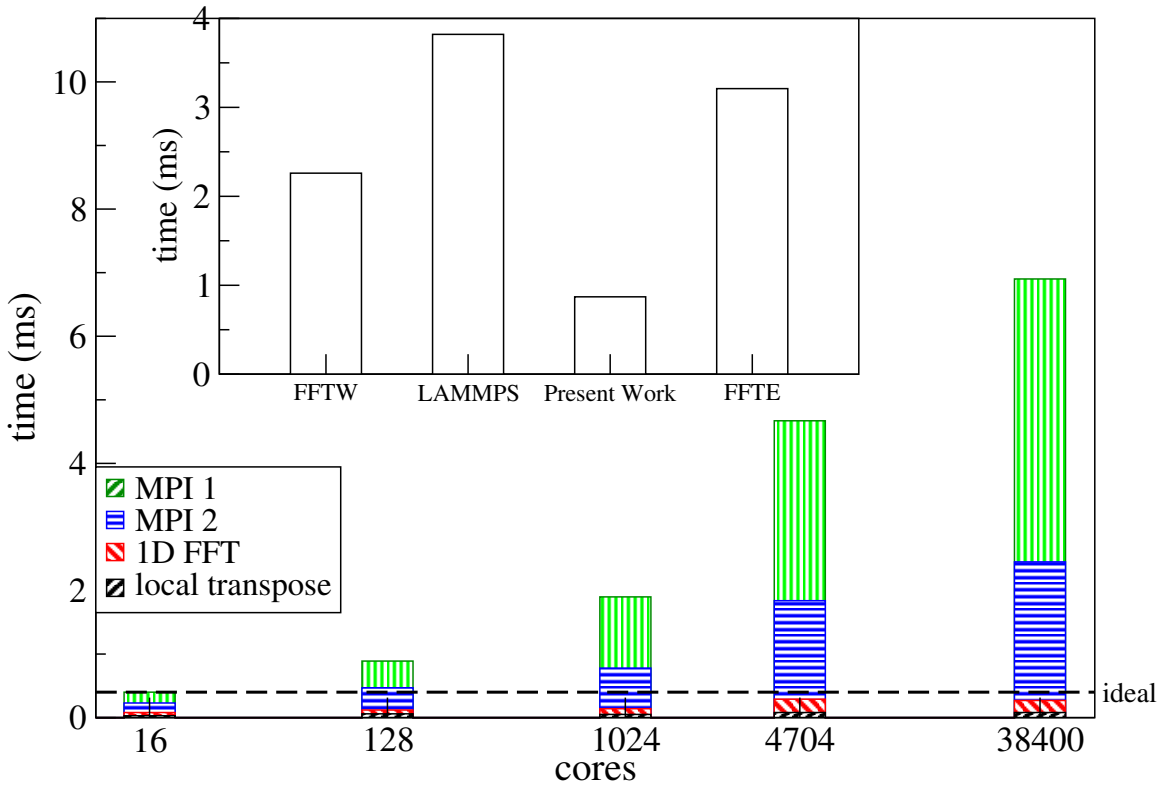


Figure 3.11: Weak scaling of complex-to-complex FFT on Cray XT5 with FFT implemented as described in appendix B.1. The 3.3 million atom system requires the 588x128x128 FFT.t_f represents the time required to compute one FFT step.

The scaling of MD codes is restricted by global communications i.e., instances when all computer nodes exchange information. Although an improvement in the FFT-part of PME speed is reported in the appendix B, MD simulations using PME still face weak-scaling problems. While for small systems, containing less than 100k atoms, simulations achieving over 100ns/day are currently possible (Hess et al., 2008; Bowers et al., 2006; Fitch et al., 2006; Freddolino et al., 2008), for larger systems the global communication for the FFT (MPI_Alltoall) takes longer than the time available for one timestep on a large number of cores. This problem worsens as the size of the system increases further, as the MPI_Alltoall global communication does not exhibit efficient weak scaling.

As shown in Figure 3.9 and Figure 3.10, the use of the RF method greatly improves the strong scaling of million-atom systems, to the point where 28ns/day are obtained when a 5.4 million atom system is run on 30k cores with a 2fs timestep. Using RF for the electrostatics calculation removes the biggest inherent limitation of the scaling of MD. While all (*i.e.*, irrespective of the method of treating the electrostatics) MD simulations in the NPT ensemble require one global communication (the MPI_Allreduce for the barostat and thermostat), this communication is not necessary at each step. The FFT part of PME, however, requires two additional global MPI_Alltoall communications, which take more time than MPI_Allreduce and do not exhibit good weak scaling, see Figure 3.11. Consequently, the performance, in ns/day, for large systems is inherently limited with PME.

The RF method has been employed in numerous studies, for example in Refs. Walser et al. (2001); Nina and Simonson (2002); Gargallo et al. (2003); Mathias et al. (2003); van der Spoel and van Maaren (2006), but has not been extensively tested for biological systems. Indeed, both the gains in computational efficiency and the possible sources of error arise from the implicit treatment of the Coulomb interaction for atoms separated more than the cutoff distance. In the present work the RF method does not appear to compromise the accuracy of MD simulation of the test system under study. This conclusion is drawn after a series of tests in

which simulations were performed with different methods for treating the long-range electrostatics interactions. The RF and Shift/Switch methods are similar in the sense that they do not consider explicitly electrostatic interactions between atoms separated by more than the cutoff distance. Consequently, one might have expected the RF and Shift methods to yield similar results. However, our findings suggest a different picture: all benchmarks show very good agreement between RF and PME, while the Shift method exhibits several significant artefacts. Also the RF and PME simulations are in very good agreement in tests on bulk water.

Tests of RF on non-neutral molecules (*e.g.*, a high charge-density protein crystal (Walser et al., 2001), a highly-charged protein domain (Gargallo et al., 2003) and RNA (Nina and Simonson, 2002)) have shown good agreement in structural properties. However, in the future, tests of dynamical properties of charged molecules would be of interest.

In the near future, it is anticipated that the performance of MD using RF might be improved to over 60ns/day for million-atom systems by using threads and asynchronous communication with neutral territory for improving parallel efficiency. In further benchmarks using the RF together with all-bond constraints and virtual sites, which allow removal of hydrogen-atom degrees of freedom enabling integration time steps up to 5fs (Hess et al., 2008), we found that 38ns/day is obtained when the 5.4-million atom system is run on 15,360 cores (data not shown). Since PME simulations are limited by the time step of the full electrostatics (*e.g.*, 6fs in the comparison), a longer time step for the short range interaction does not improve the performance of PME as it would for RF. Thus for large systems, a significant improvement in performance by employing longer time steps is more easily achieved using RF for the electrostatics. However, we stress that carefully-designed benchmarks should first be performed before 5fs-timestep simulations are routinely applied to biomolecular systems.

Some critical biological phenomena, such as ligand binding and the folding of small proteins, require the simulation of relatively small systems (*e.g.*, $\sim 10^4$ atoms

or ∼1-10nmlength scales) for relatively long timescales (*e.g.*, 10^3s). For this type of application the strategy described here is not applicable. Rather, the present approach permits efficient atomistic MD simulation of larger, multimillion-atom biomolecular systems (*i.e.*, on a length scale ∼100nm) for times of ∼30ns/day. Using the proposed strategy simulations of these large systems for timescales approaching the microsecond would now seem to be within reach on the Cray XT5. We anticipate that a wealth of structural and dynamical information of biological importance will thus be revealed.

Chapter 4

Simulation Analysis of the Temperature Dependence of Lignin Structure and Dynamics

This chapter is revised based on a paper with the same title as the chapter published in J. Am. Chem. Soc., 2011, 133(50), 20277–20287 authored by Petridis, L.; Schulz, R.; and Smith, J. C. My primary contributions to this paper include (i) deciding the simulation approach (ii) running the simulations (iii) computing the structural properties of lignin, and (iv) computing the entropy of water.

Abstract

Lignins are hydrophobic, branched polymers that regulate water conduction and provide protection against chemical and biological degradation in plant cell walls. Lignins also form a residual barrier to effective hydrolysis of plant biomass pretreated at elevated temperatures in cellulosic ethanol production. Here, the temperature-dependent structure and dynamics of individual softwood lignin polymers in aqueous solution are examined using extensive (17μs) molecular dynamics simulations. With decreasing temperature the lignins are found to transition from mobile, extended to glassy, compact states. The polymers are comprised of blobs, inside which

the radius of gyration of a polymer segment is a power-law function of the number of monomers comprising it. In the low temperature states the blobs are inter-permeable, the polymer does not conform to Zimm/Stockmayer theory, and branching does not lead to reduction of the polymer size, the radius of gyration being instead determined by shape anisotropy. At high temperatures the blobs become spatially separated leading to a fractal crumpled globule form. The low-temperature collapse is thermodynamically driven by the increase of the translational entropy and density fluctuations of water molecules removed from the hydration shell , thus distinguishing lignin collapse from enthalpically driven coil-globule polymer transitions and providing a thermodynamic role of hydration water density fluctuations in driving hydrophobic polymer collapse. Although hydrophobic, lignin is wetted, leading to locally-enhanced chain dynamics of solvent-exposed monomers. The detailed characterization obtained here provides insight at atomic detail into processes relevant to biomass pretreatment for cellulosic ethanol production and general polymer coil-globule transition phenomena.

4.1 Introduction

Lignins are branched hydrophobic heteropolymers that provide mechanical strength to plant cell walls, regulate water conduction and play an important role in plant defense against enzymatic or microbial degradation (Grabber, 2005; Jorgensen et al., 2007). Lignins are also of central interest in biofuels research. Although plant biomass has the potential to be a renewable feedstock for industrial biofuel production, its recalcitrance to hydrolysis, which arises in part from lignin, necessitates expensive pretreatment prior to fermentation that significantly increases cost (Himmel et al., 2007). Pretreatment technologies commonly employed increase biofuel yield by disrupting the lignin-carbohydrate network that physically prevents enzymes from

reaching and hydrolyzing cellulose (Yang and Wyman, 2008). A variety of pre-treatment methods exist, the most common of which, such as dilute-acid, ammonia-based and hydrothermal, involve temperatures $\gtrsim 100°C$ (Yang and Wyman, 2008). Understanding the temperature dependence of lignin structure and dynamics is thus of particular importance to next-generation biofuel production.

Lignin is known to be "hard" and glassy at room temperature and to "soften" above its glass transition temperature, which ranges between $80-100°C$ for softwoods (Irvine, 1985; Bjurhager et al., 2010). Temperature influences the quality of water as a solvent for polymers: whereas at high temperatures water can be a "good" solvent, leading to the tendency of polymers to assume extended chain conformations, as the temperature decreases, water can become a "poor" solvent and polymers tend to collapse to dense globules. This polymer coil-globule transition has been studied extensively experimentally (Sun et al., 1980; Stepanek et al., 1982; Kubota et al., 1990; Wu and Wang, 1998), theoretically (Stockmayer, 1960; Flory, 1966; DeGennes, 1975) and with simulation (Ma et al., 1995; Zhou et al., 1997; Polson and Zuckermann, 2002; Steinhauser, 2005).

The collapse at low temperatures is attributed to the hydrophobic effect, an interaction that drives diverse biological self-assembly phenomena, such as protein folding and membrane formation (Chandler, 2005; ten Wolde and Chandler, 2002; ten Wolde, 2002; Athawale et al., 2007; Miller et al., 2007; Li and Walker, 2010). Hydrophobicity of idealized solutes, *i.e.* that do not exert attractive interactions with water, manifests itself differently on small and large length scales (Chandler, 2005; Lum et al., 1999; Berne et al., 2009). The free energy cost of solvating small idealized hydrophobic particles is entropic, driven by the formation of small water cavities that accommodate the solute without destroying water hydrogen bonds (Hummer et al., 1996; Garde et al., 1996). In contrast, large hydrophobic solutes break water-water hydrogen bonds, leading to the creation of a liquid-vapor interface and a large positive enthalpy of solvation (Berne et al., 2009; Stillinger, 1973).

Water in the hydration layer of biomolecules has been extensively studied. Near proteins water has been shown to be denser than the bulk (Svergun et al., 1998; Merzel and Smith, 2002) and to exhibit slower dynamics (for a review see Ref. 31). Simulations have also found the sub-nanosecond translational and rotational entropy of hydration water molecules near polyamidoamine dendrimers (Lin et al., 2005), lipid bilayers (Debnath et al., 2010) and DNA (Jana et al., 2006) to be lower than the bulk. However, the structural and thermodynamic properties of lignin and its hydration water have so far not been investigated.

To our knowledge, there have been to date only two previous MD simulation studies of lignin. The first examined the organization of lignin oligomers on a cellulose surface. Due to computational limitations these simulations were limited to short trajectories performed in vacuum (Besombes and Mazeau, 2005). More recently, a combination of small angle neutron scattering and MD simulation examined the surface morphology of softwood lignin aggregates, which were found to exhibit self-similar surface properties constant over three orders of magnitude in length (Petridis et al., 2011).

Here, the effect of temperature and branching on the structure and dynamics of individual softwood lignin molecules in aqueous solution is investigated with the use of extensive ($17.5\mu s$) atomistic molecular dynamics simulations. Due to their highly aggregating nature, individual lignin polymers can be found only in very dilute aqueous solutions, thus presenting steep challenges for their experimental characterization (Islam et al., 2000). However, thermochemical pretreatment dissociates plant-cell wall lignin, and, due to their hydrophobic character, the residual lignin polymers then collapse and coalesce to form clumps (Petridis et al., 2011; Pingali et al., 2010; Selig et al., 2007; Donohoe et al., 2008; Kristensen et al., 2008; Chundawat et al., 2011). The computational investigation of the behavior of lignin polymers as a function of temperatures can provide insight into collapse and coalescence processes relevant to pretreatment of plant biomass.

The paper presents a comprehensive analysis of the temperature dependence of the size, shape, scaling properties, hydration and dynamics of softwood lignin with varying degress of branching. The lignin polymers are found to transition from high-temperature mobile and extended states to glassy and compact states at low temperatures. Hydration shell entropy is found to be at the thermodynamic origin of the lignin collapse at low temperatures.

4.2 Methods

4.2.1 Model Systems

Structural models of individual lignin molecules were generated by using available experimental information on the average chemical composition of softwood lignins (Petridis et al., 2011). Softwood lignins are composed primarily of guaiacyl (G) monomers connected by various linkages, leading to the formation of branched and unbranched biopolymers (Pu et al., 2008). Here, nine different lignin molecules were simulated.

Each molecule comprised 61 G units, with a molecular weight of $\sim 13kDa$, within the experimentally-determined range (Brunow et al., 1993). The average interunit-linkage composition was that of softwoods: β-O-4' 50%, 5-5' 30%, α-O-4' 10% and β-5' 10% (Pu et al., 2008). The number of branch points and their location along the chain were assigned randomly using a computer algorithm: two lignins, termed $L0_a$ and $L0_b$, have zero branch points, lignins $L1_a$ and $L1_b$ have one, $L2$ two, $L3$ three, $L4$ four, $L5$ five and $L6$ six. This distribution is consistent with the experimentally-determined average linkage density for spruce wood, which is 0.052, or 3.2 branch points per 61 monomers (Yan et al., 1984).

Hence, the primary structures of the nine lignins simulated here are different from each other but consistent with the average chemical composition of softwood lignin. For example, although for all molecules 50% of the linkages are of the β-O-4' kind,

the position of these linkages along the chains varies between molecules, as does the position of the branch points and the lengths of the branches. The sequence of linkages of each lignin can be found in Tables C.1-C.9 of appendix C.

The resulting nine lignin molecules were subsequently individually solvated and subject to molecular dynamics simulations. Each lignin polymer was solvated in a rhombic dodecahedron with an inscribed sphere radius of 52.5Å containing in total $\sim$ 81600 atoms.

4.2.2 Molecular Dynamics Simulation Details

The CHARMM force field for lignin (Petridis and Smith, 2009) and the TIP3P water model (Jorgensen et al., 1983) were employed. Periodic boundary conditions were employed and the PME algorithm (Darden et al., 1993; Essmann et al., 1995) was used for electrostatic interactions. A reciprocal grid of 80 x 80 x 80 cells (84x84x84 for 480K) was used with 4th order B-spline interpolation. A cut-off of 12Å was used for the neighbor searching and real-space electrostatics. Charge groups were used for water but not for lignin. For the van der Waals interactions the switch function was used for distances $9 - 10$Å.

The simulations were performed with the program GROMACS 4.5.1 (Hess et al., 2008; van der Spoel et al., 2005a; Lindahl et al., 2001; Berendsen et al., 1995b) using a time step of 2fs. Atoms involving hydrogens were constrained using the LINCS (Hess, 2008) algorithm (4th order with one iteration) and for water the Settle algorithm was used (Miyamoto and Kollman, 1992). Neighbor searching was performed every 10 steps. Temperature coupling was performed with the V-rescale algorithm (Bussi et al., 2007) ($\tau = 0.1$fs) and pressure coupling with the Parrinello-Rahman algorithm (Parrinello and Rahman, 1981) ($\tau = 1$fs).

For the four lignins with either one or zero branch points, each system was first heated from 0K to 480K in 1ns and subsequently simulated at 480K for a total of 100ns. Ten structures were taken from this trajectory at 10ns, 20ns, ..., 100ns.

Starting from these ten structures each of the four lignins was simulated at four different temperatures: 300K, 360K, 420K and 480K. The cooling from 480K to the target temperature was performed in 5ns, then the simulation was equilibrated for 55ns, followed by 50ns of production used for the analysis. The above protocol was chosen in order to obtain ten very different starting structures, since preliminary simulations had shown that at 480K large structural changes take place on the 100ns timescale.

The five separate MD simulations for each of the $L2$, $L3$, $L4$, $L5$ and $L6$ lignins were performed at 300K. The simulations were equilibrated for 55ns, followed by 50ns of production used for the analysis.

The total simulation length was $17.5\mu s$. All simulations were performed on the Jaguar Cray XT5 supercomputer at Oak Ridge National Laboratory.

4.2.3 Analysis of Molecular Dynamics Simulation

Structural Properties of Lignin

The radius of gyration R_g and the gyration tensor were computed using GROMACS g_polystat using all the lignin atoms and without mass-weighting. The asymmetry of the chain conformations was described by the asphericity, Δ,

$$\Delta = \left\langle \frac{(L_1 - L_2)^2 + (L_2 - L_3)^2 + (L_1 - L_3)^2}{2(L_1 + L_2 + L_3)^2} \right\rangle, \tag{4.1}$$

where L_1, L_2 and L_3 are the eigenvalues of the radius of gyration tensor of the particular lignin molecule and $< \ldots >$ represents an ensemble average, replaced here by a time average. The shape of the molecule was further characterized by calculating the prolateness, Σ

$$\Sigma = \left\langle \frac{(2L_1 - L_2 - L_3)(2L_2 - L_3 - L_1)(2L_3 - L_1 - L_2)}{2(L_1^2 + L_2^2 + L_3^2 - L_1L_2 - L_2L_3 - L_3L_1)^{3/2}} \right\rangle. \tag{4.2}$$

For a solid ellipsoid with unit mass density and radius of gyration $r_{g,i}$ around axis i it follows (Lamb, 1920)

$$r_{g,i}^2 = \frac{1}{5}\sum_{j\neq i} r_j^2 = \sum_{j\neq i} L_j \Rightarrow r_i = \sqrt{5L_i} \tag{4.3}$$

where r_i is the radius and L_i is the eigenvector of the gyration tensor introduced before. Thus the volume of the ellipsoid is

$$V = \frac{4}{3}\pi r_1 r_2 r_3 = \frac{4}{3}\pi 5^{3/2}\sqrt{L_1 L_2 L_3} \tag{4.4}$$

The solvent accessible surface area (SASA) was computed using GROMACS g_sas with a probe radius of 1.4Å. The van der Waals radii for the lignin atom types were set to C: 1.5Å, H: 1Å, and O: 1.3Å. A lignin atomic contact was defined as any pair of lignin atoms separated by < 3Å. Hydrogen bonds were defined by a < 3.5Å donor-acceptor distance and a donor-H-acceptor angle between 150° and 210°.

Lignin Dynamics

Monomer mean square displacements,

$$< \Delta r_n^2 >= \left\langle [r_n(t) - r_n(0)]^2 \right\rangle, \tag{4.5}$$

where $r_n(t)$ is the position of the n^{th} monomer at time t, were calculated after aligning each trajectory frame to the preceding frame to remove whole lignin molecule translation and rotation. Data for calculating $< \Delta r_n^2 >$ were collected over the last 50ns of 40 MD 100ns trajectories (four molecules simulated 10 times each).

Structure of Hydration water

The water molecules were classified into (a) "hydration shell" water, defined by a 4.9Å distance cutoff between water oxygen atoms and any lignin non-hydrogen atom, where

this cutoff distance was determined as the minimum in the proximal distribution function and (b) "bulk" water, defined by a distance of water oxygen atom to all lignin heavy atoms > 4.9Å.

The proximal distribution function $g_{prox}(r)$ is given by (Ashbaugh et al., 2005):

$$g_{prox} = \frac{\langle n \rangle}{A(r)\Delta r \rho_{bulk}}, \tag{4.6}$$

where $\langle n \rangle$ is the average number of water oxygen atoms found at a distance $[r, r + \Delta r]$ from a non-hydrogen atom on the surface of the lignin (here $\Delta r = 0.1$Å), $A(r)$ is the SASA of the lignin calculated with a probe radius r and $\rho_{bulk} = 0.0327\text{Å}^{-3}$, 0.0307Å^{-3}, 0.0280Å^{-3} and 0.0242Å^{-3} are the bulk number-densities of the TIP3P water model determined from four separate pure water simulations at $T = 300$K, 360K, 420K and 480K respectively. The product $A(r)\Delta r$ is the approximate volume of the shell of water molecules whose distance to the lignin surface is between r and $(r + \Delta r)$Å.

The isothermal compressibility, χ, is a measure of the density fluctuations of water in the grand canonical ensemble (Dadarlat and Post, 2001):

$$\chi = \frac{V}{k_B T} \frac{\langle W^2 \rangle - \langle W \rangle^2}{\langle W \rangle^2} \tag{4.7}$$

where W is the number of water molecules in (fixed) volume V. The compressibility of the lignin hydration water, χ_{hydr} was calculated by counting the number of water molecules in the hydration shell as a function of time. To derive a value of χ_{bulk}, the compressibility of bulk water, that can be directly compared to χ_{hydr}, the lignin surface was superimposed onto the results of a control simulation of pure water (Merzel and Smith, 2002), and the number of water molecules inside the hydration shell volume was calculated as a function of time. Values of both χ_{hydr} and χ_{bulk} were derived from six 40ps simulations: in three of these lignin was in extended states and in the other three lignin was collapsed, see Figure C.1 of appendix C.

Entropy of Water

A two-phase thermodynamic model was employed that partitions the translational and rotational density of states of water molecules $g(\omega)$ into gas-like, $g^g(\omega)$, and solid-like, $g^s(\omega)$ components (Lin et al., 2003, 2010):

$$g(\omega) = g^g(\omega) + g^s(\omega) \tag{4.8}$$

$$= \frac{2}{k_B T} \lim_{\tau \to \infty} \int_{\tau}^{\tau} C(t) e^{-i\omega t} dt, \tag{4.9}$$

where $g(\omega)$ is the Fourier transform of the velocity-autocorrelation function (VACF) $C(t)$. $C(t)$ is either the mass weighted VACF of the center of mass velocities:

$$C_T(t) = \sum_{i=1}^{W} m \langle \overrightarrow{v}_i(t) \cdot \overrightarrow{v}_i(0) \rangle, \tag{4.10}$$

or the moment of inertia weighted angular VACF

$$C_R(t) = \sum_{j=1}^{3} \sum_{i=1}^{W} \langle I_j \omega_{ij}(t) \omega_{ij}(0) \rangle. \tag{4.11}$$

W is the number of water molecules in the system, m is the mass, $\overrightarrow{v}_i(t)$ the velocity of the i^{th} water molecule at time t, I_j and ω_{ij} are the j^{th} principal moment of inertia and angular velocity of water molecule, i. The translational entropy of water can be calculated by assigning the appropriate weight, λ, to the gas- and solid-like components:

$$S = \frac{1}{2\pi} \int_0^{\infty} g^g(\omega) \lambda^g(\omega) + \frac{1}{2\pi} \int_0^{\infty} g^s(\omega) \lambda^s(\omega). \tag{4.12}$$

Details on the decomposition of $g(\omega)$ and the derivation of the weighting functions W^g and W^s can be found in appendix C. $C(t)$ (Equation 4.10) for bulk water was determined from ten pure water 20ps simulations, where velocities and coordinates were saved every 4fs. $C(t)$ for the hydration-shell water molecules was determined

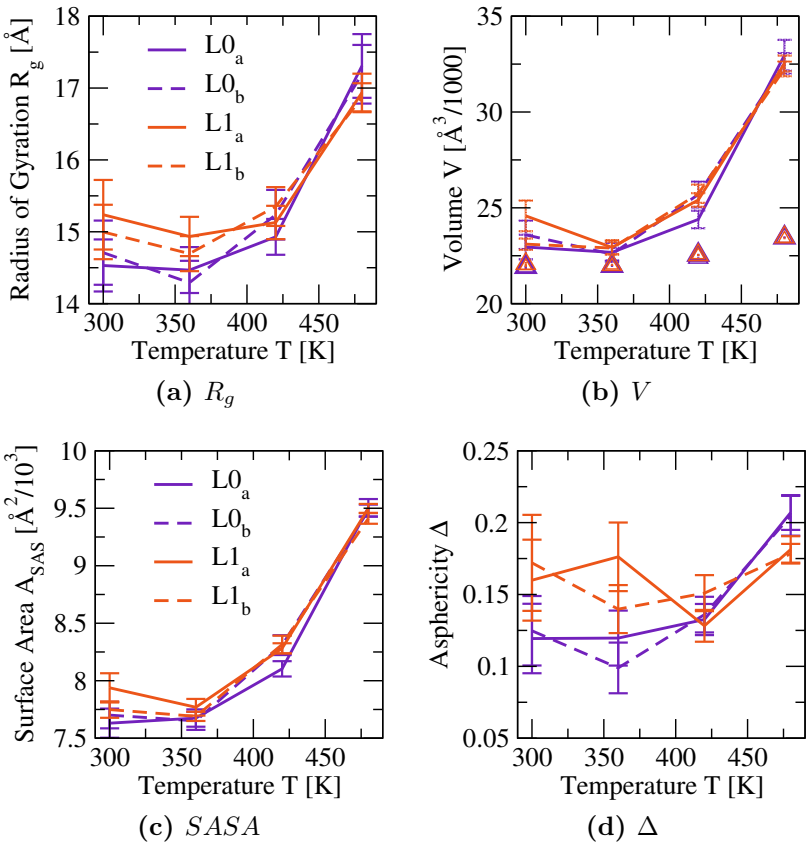
from ten lignin-water 20ps simulations, each starting from a different lignin configuration, from which data were saved every 4fs. The sum in Equations 4.10 and 4.11 was taken over hydration-shell water molecules only.

4.3 Results

4.3.1 Structure

The radius of gyration, R_g, (Figure 4.1a) for all the lignin molecules exhibits a strong temperature dependence above $T \approx 420$K with $R_g \approx 15$Å for $T \leq 420$K and $R_g \approx 17$Å for $T = 480$K. Figure 4.2 graphically represents the polymer chain at 300K and 480K. The probability distribution of R_g at $T = 480$K not only has a higher mean, but is also considerably broader than at $T = 300$K (Figure 4.3a) thus allowing the chain to increase entropy by exploring more conformations at high temperatures. Conversely, the highest number of intermolecular lignin contacts is found in the low temperature collapsed state (Figure 4.3b). Surprisingly, lignin also makes the highest number of hydrogen bonds to water at low temperatures (Figure 4.3b, orange), although possessing the lowest solvent accessible surface area (SASA) (Figure 4.1c). As discussed later, this is due to a higher hydration-water density at low temperatures.

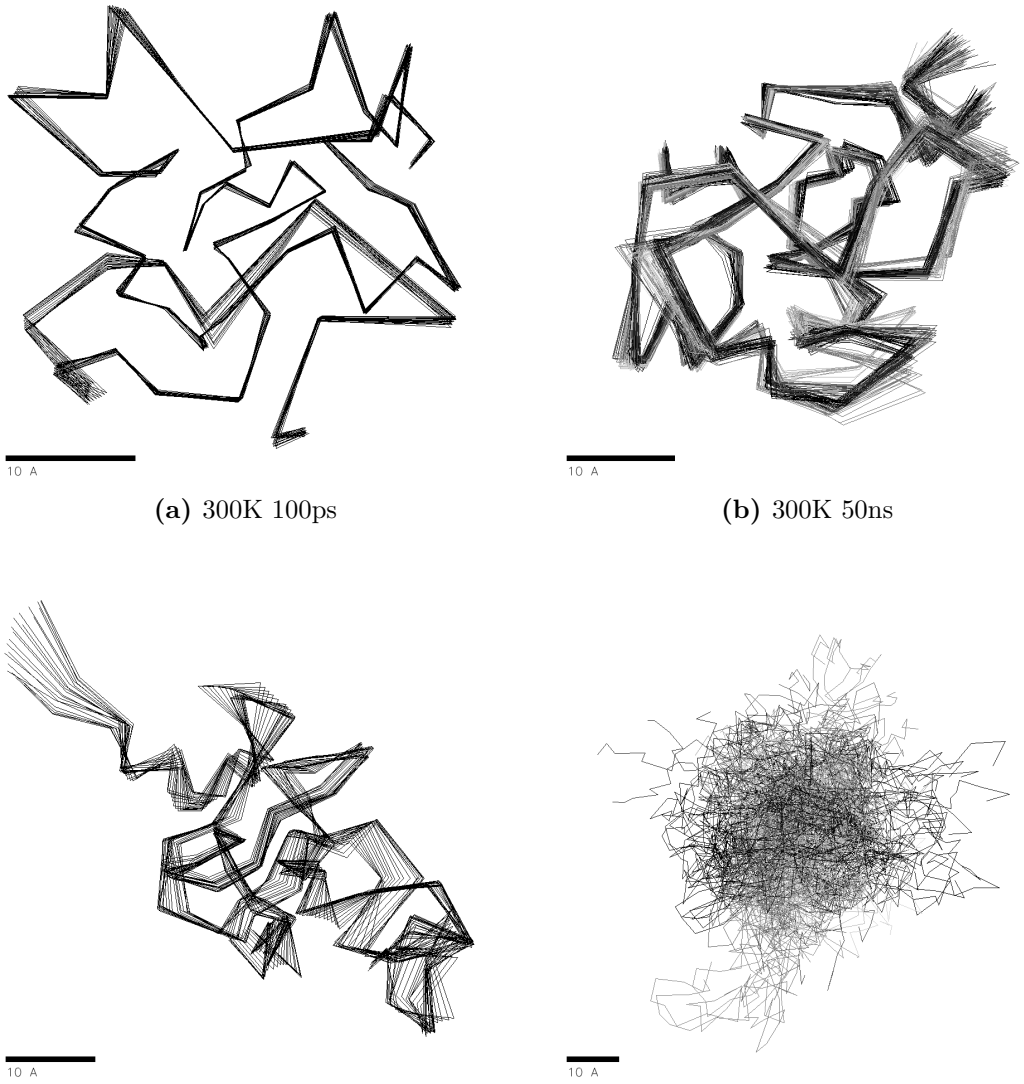
Figure 4.1b shows the volume within the SASA, V_{SAS}, and the volume of the gyration ellipsoid, V_{gyr}. No difference between branched or unbranched lignins is seen for either of these quantities. At the two lower temperatures (300K and 360K) the volume of the gyration ellipsoid is a good approximation to V_{SAS}, overestimating the volume by approximately the error bar. However, for the two higher temperatures (420K and 480K) the gyration ellipsoid overestimates V_{SAS} significantly. The fact that V_{gyr} does not vary significantly with temperature excludes the presence of cavities at high temperatures. Therefore, the increase in R_g and V_{SAS} can be understood by the polymer adopting more extended conformations.



(a) R_g **(b)** V

(c) $SASA$ **(d)** Δ

Figure 4.1: Temperature dependence of structural properties of unbranched (L0) and branched (L1) lignins. (a) Radius of gyration R_g (b) SASA volume (V_{sas}, triangles) and gyration ellipsoid volume (V_{gyr}, lines) (c) SASA (d) asphericity Δ defined by Equation 4.1. Each data point is derived from ten trajectories. The average values of R_g, V_{gyr} and $SASA$ of each trajectory were first computed and the means of these ten average values are listed, with the error estimated as $\sigma\sqrt{9}$, where σ is the standard deviation of the mean. For V_{SAS} the values of the four molecules mostly overlap and the error estimate is not shown because it is similar to the line thickness.



(a) 300K 100ps

(b) 300K 50ns

(c) 480K 100ps

(d) 480K 50ns

Figure 4.2: Structure of $L0_a$ at 300K and 480K. Shown are lines connecting the center of mass of each residue. (a) and (c) are the structures obtained from a 100ps trajectory saved every 5ps (smoothed over 5 frames). Structures in (b) and (d) were obtained from a 50ns trajectory saved every 500ps (smoothed over 100 frames). Time is mapped using a black (beginning of trajectory) to white (end of trajectory) scale.

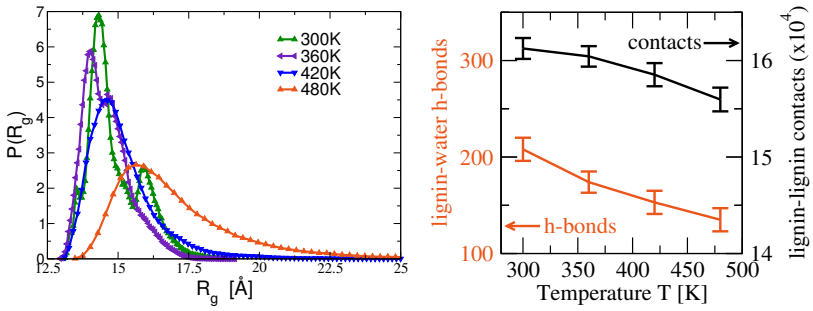


Figure 4.3: Temperature dependence of (a) the probability distribution of the radius of gyration, R_g (b) the lignin intra-molecular contacts (black; squares) and lignin-water hydrogen bonds (orange; diamonds). Data represent ensemble averages of L0 and L1.

4.3.2 Scaling Properties

The scaling concept of polymers (Gennes, 1979) is now employed to examine the size of a linear segment of lignin polymers comprising $(N+1)$ monomers. For the ensemble of the two unbranched lignins at $T = 300$K, the radius of gyration of the corresponding segment (denoted as $r_g(N)$ so as to distinguish it from $R_g = r_g(N_{tot})$ the radius of gyration of a whole polymer) is found to follow a power-law behavior for $N \lesssim N_c = 30$ (Figure 4.4a):

$$r_g \propto N^{\nu}, \tag{4.13}$$

where the scaling exponent $\nu = 0.34 \pm 0.01$. The power-law indicates self-similar packing density for short segments ($N \lesssim N_c$). For larger segment lengths, $N \gtrsim N_c$, the size of the segments increases more slowly with length and r_g approaches a plateau $r_g \sim N^0$.

The flattening of $r_g(N)$ for long segments ($N > N_c \simeq 30$) is a feature of linear homopolymers comprised of "blobs" (Ma et al., 1995). Equation 4.13 is valid inside these blobs, whose segment length is $\sim N_c$ (Ma et al., 1995). Blobs can penetrate each other giving rise to the plateau $r_g \sim N^0$ for low temperatures. At 300K the blobs are in close spatial proximity and the average distance, D between monomers belonging to the same blob, is comparable to D between monomers on different blobs (Figure 4.4b, compare $N \simeq 11$ with $N \simeq 41$). The crossover length, N_c can be determined by the condition that the size of the blob is of the order of the polymer size, *i.e.* $N_c \approx (R_g/a)^3 = 31$, where $a = 4.6$Å is the radius of gyration of a single monomer (Ma et al., 1995).

While $r_g(N)$ at $T = 300$K, 360K and 420K displays the crossover at N_c, at the highest temperature ($T = 480$K) the power-law behavior of Equation 4.13 is observed over the entire range of N (Figure 4.4a). This self-similar fractal spatial chain structure, called a "crumpled globule", arises when the "blobs" are spatially segregated from each other (Grosberg et al., 1988, 1993). The key difference between the crumpled and low-T globules is that, in the former, monomers distant along the

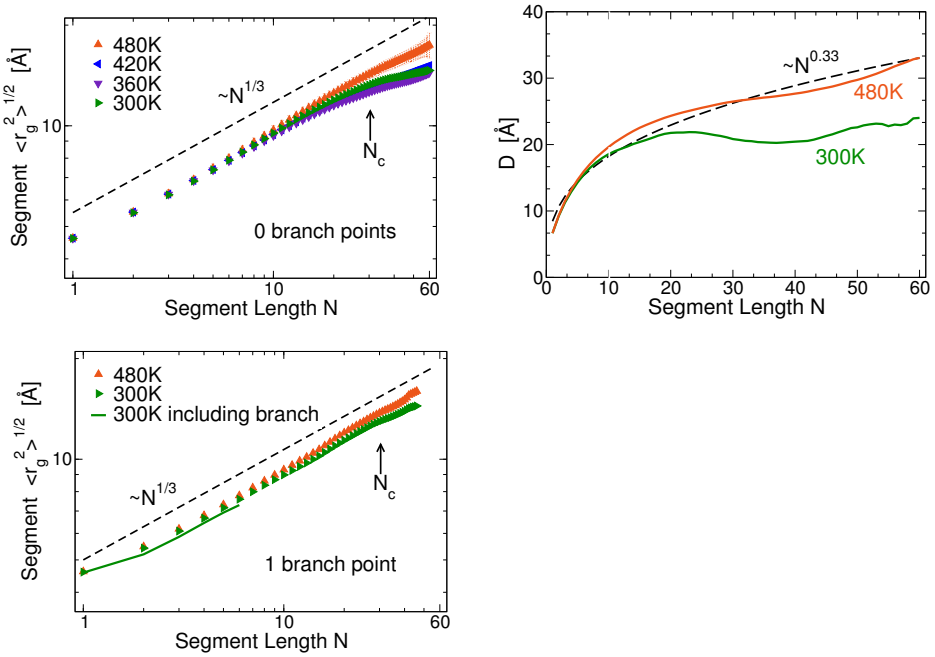


Figure 4.4: Scaling properties of the unbranched lignins at different temperatures. (a) Root mean square of the radius of gyration $r_g(N)$ of a polymer segment comprising $(N+1)$ monomers. (b) Distance D between monomers i and j as a function of $N = |i-j|$. (c) $< r_g^2(N) >^{\frac{1}{2}}$ of the ensemble of polymers with one branch point. Also shown, as a green solid line, are data from short ($(N \leq 6)$) segments that include the branch point. In all plots, the dashed black line is a $\sim N^{0.33}$ power-law function and all plots are time averages over the last 50ns of the ensemble of 20 MD trajectories. The error bars are the standard deviations of the ensemble distribution.

chain are distant in space, whereas for the latter monomers distant along the chain have a relatively high probability of being proximal in space. Recently, chromatin was also found to exist in crumpled globules (Lieberman-Aiden et al., 2009).

The ensemble of the linear segments of the two branched lignins (called $L1$) also displays the power-law dependence of Equation 4.13 for $N \lesssim N_c$, with an exponent $\nu = \nu_{L1} = 0.033 \pm 0.01$, and a crossover behavior for $N \gtrsim N_c$ at low temperatures (Figure 4.4c). $r_g(N)$ was also computed for the ensemble (called $L1_{br}$) of short ($N \leq 6$) segments that contain the branch point, either on the ends of the segment or in its interior. $r_g(N)$ of $L1$ and $L1_{br}$ are statistically indistinguishable and therefore the presence of the branch point does not alter the scaling behavior of collapsed lignins.

4.3.3 Effect of Branching

Lignins are randomly branched and it is therefore of interest to examine how branching affects the size and shape of the polymers. The ratio, $g = \langle R_g^2 \rangle_{L1} / \langle R_g^2 \rangle_{L0}$, of the mean square radius of gyration of the branched polymer ($L1$) to that of a linear polymer having the same molecular weight ($L0$) quantifies the effect of branching on the polymer size.

For polymers in a good solvent, branched chains assume a smaller R_g than their linear counterparts of the same molecular weight (Freire, 1999; Wang et al., 2004). The theory by Zimm and Stockmayer (ZS) (Zimm and Stockmayer, 1949) predicts $g < 1$ for various branching configurations of polymers in ideal solvents with the assumption of isotropy, and has been employed successfully to interpret experimental data *e.g.* (Wang et al., 2004; Yu et al., 2005). In appendix C we modify the ZS theory to render it applicable to collapsed polymers in bad solvents and demonstrate that, while the decrease in polymer size due to branching is predicted to be smaller for bad solvents than for ideal solvents, g remains < 1. For example, with the modified ZS for a star polymer of three equal-length arms, $g = 0.86$ for a bad solvent compared to $g_{ZS} = 0.77$ for an ideal solvent.

Interestingly, the present simulations show the unbranched lignins to mostly have smaller R_g than the branched lignins. Hence, for lignin at 300K $g > 1$. This is because, in contrast to the assumption of anisotropy made in the analytical theory, the simulated lignins are not spherical. The anisotropy of the chain conformations can be described by the asphericity Δ (Equation 4.1), that takes values between $\Delta = 0$, corresponding to a spherical shape, and $\Delta = 1$, for a rod-like shape. Table 4.1 offers an explanation of the trend in the calculated radii of gyration: the more aspherical the molecule, the higher its R_g (see also Figure C.2 in appendix C).

The shape of a polymer is further characterized by the prolateness Σ (Equation 4.2). Most of the present lignins have a prolate "melon-like" configuration with $\Sigma > 0$, the exception being molecule $L0a$, which adopts both prolate ($\Sigma > 0$) and oblate "disk-like" ($\Sigma < 0$) configurations.

The temperature dependence of R_g (Figure 4.1a) is different for the branched and unbranched lignins, such that at low T the R_g is higher for the branched molecules ($g > 1$) whereas for high T it is higher for the unbranched lignins ($g < 1$). This difference in the temperature trend between the $L0$ and $L1$ lignins correlates with the asphericity, Δ shown in Figure 4.1d. Hence, at low T the branch points lead to increased asphericity and thus a larger R_g . The asphericity for the branched lignins has no significant temperature dependence. For both the branched and unbranched lignins the temperature dependence is smaller than that that would be expected for a constant-mass ellipsoid the R_g of which displays the temperature dependence of Figure 4.1a.

4.3.4 Structure of Hydration Water

Structural properties of water close to the surface of the lignin are quantified by the proximal distribution function $g_{prox}(r)$, given by Equation 4.6. At low temperatures $g_{prox}(r)$ is significantly structured, displaying two peaks at 2.8Å and the other at 3.5Å (Figure 4.5a). This double-peak feature of the first hydration shell has been

Table 4.1: Structural properties of the lignin molecules with various number of branch points (indicated by the number in the molecule name): R_g is the radius of gyration, Δ the asphericity (Equation 4.1) and Σ the prolateness (Equation 4.2) at $T = 300$K. Quantities are averaged over the last 50ns of the simulations. For molecules with zero and one branch points, the average value of R_g, Δ and Σ of each trajectory was first computed and the mean of these ten average values is listed, with the error estimated as $\sigma\sqrt{9}$, where σ is the standard deviation of the mean. For molecules with more than two branch points the error is estimated from the standard deviation of the averaging of the single trajectory.

name	R_g	$< \Delta >$	$< \Sigma >$
$L0_a$	14.0±0.1	0.08±0.01	−0.25±0.23
$L0_b$	15.5±0.3	0.15±0.03	0.49±0.16
$L1_a$	14.4±0.4	0.08±0.03	0.51±0.16
$L1_b$	15.2±0.8	0.12±0.03	0.02±0.30
$L2$	16.0±0.6	0.20±0.04	0.65±0.06
$L3$	15.1±0.5	0.12±0.04	0.25±0.35
$L4$	14.4±0.2	0.10±0.02	0.30±0.11
$L5$	17.6±0.3	0.20±0.02	0.11±0.21
$L6$	14.0±0.2	0.06±0.01	0.49±0.28

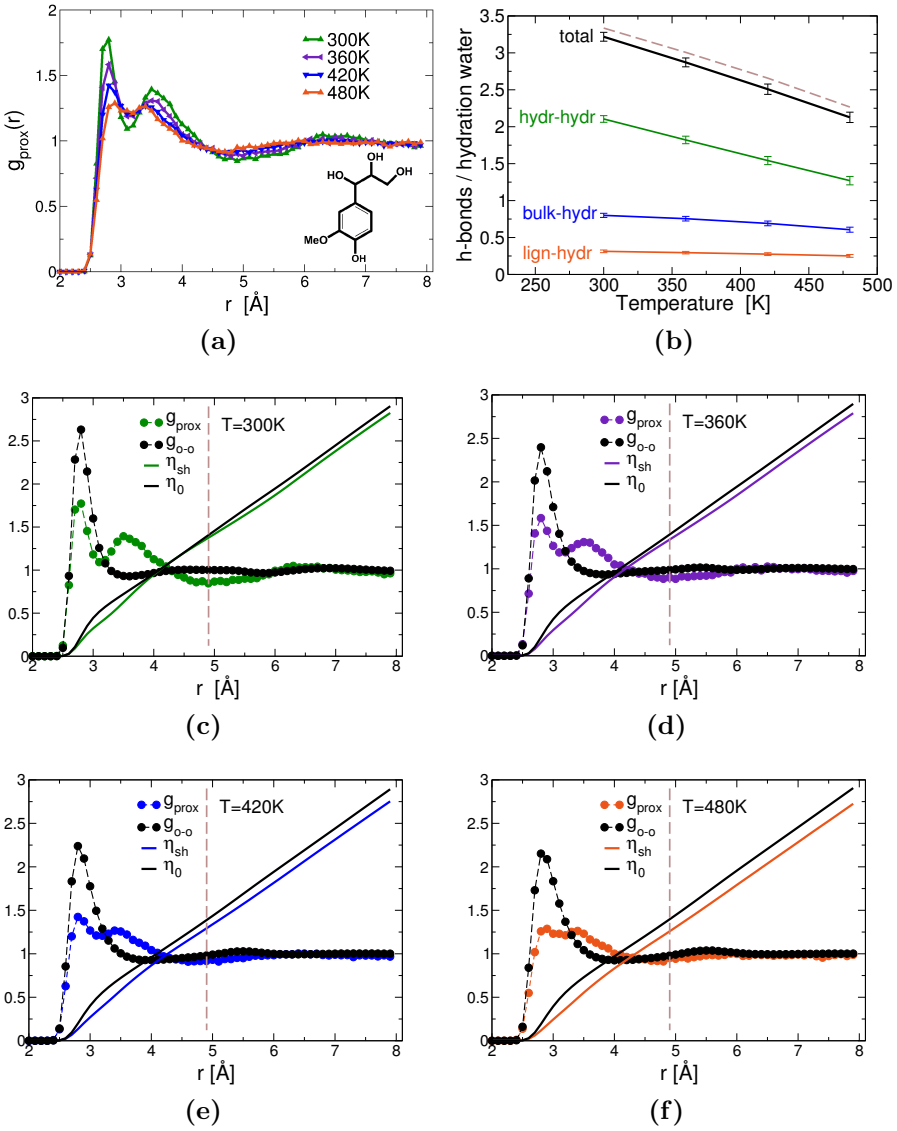


Figure 4.5: (a) Proximal distribution function of water oxygen atoms at a distance r from the surface of the lignin. Data are averaged over the last 50ns of the 10 MD trajectories of the $L0_a$ lignin with no branch points. Inset: Chemical structure of a guiacyl monomer. (b) Average number of hydrogen bonds a hydration water molecule makes with other hydration-shell waters, bulk water and lignin. The dotted brown line is the number of h-bonds per bulk water molecule. (c-f) Proximal distribution function of the oxygen atoms of lignin hydration water (g_{prox}), radial distribution function of bulk water oxygen atoms (g_{o-o}) and the respective cumulative sums, η_{sh} and η_0. The dashed vertical line, $r = 4.9$Å, marks the outer boundary of the hydration shell of lignin.

previously reported for the hydration of larger lignin aggregates (Petridis et al., 2011) and proteins (Levitt and Sharon, 1988; Smolin and Winter, 2004), and arises from the first hydration shells of the polar oxygen atoms at ~ 2.8Å and non-polar carbon atoms at ~ 3.5Å.

Lignin contains both polar hydroxyl and non-polar aliphatic and aromatic groups, see the inset of Figure 4.5a. For the ensemble of nine lignins at $T = 300$K, with various degrees of branching, the average fraction of the SASA which is hydrophilic is $\langle\phi\rangle_{pol} = 0.43 \pm 0.01$, where a "hydrophilic" atom is crudely defined as having partial charge $|q| > 0.2e$. This is higher than the average fraction of the surface area of an isolated monomer that is hydrophilic: $\langle\phi\rangle_{mon} = 0.37 \pm 0.01$. $\langle\phi\rangle_{pol} > \langle\phi\rangle_{mon}$ indicates that hydrophilic hydroxyl moieties of lignin are preferentially exposed to the solvent in order to maximize favorable interactions with the water molecules. In a separate 80ns MD simulation of lignin $L0_a$ in vacuum the behavior was different due to burial of hydrophilic groups, with $\langle\phi^{vac}\rangle_{pol} = 0.36 \pm 0.01$ similar to $\langle\phi\rangle_{mon} = 0.37$.

Temperature increase leads to gradual loss of local hydration layering, as shown by the decrease of the hydration peaks in Figure 4.5a. When comparing the low- and high-temperature simulations, it is observed that the local water density around the polar lignin atoms decreases more than around the nonpolar atoms and that the position of the nonpolar peak shifts to a slightly smaller value of r. However, $\langle\phi\rangle_{pol} = 0.43$ for all temperatures, and so lignin collapse at low T is not associated with decreased exposure of hydrophobic moieties.

The density of the lignin hydration shell, ρ_{sh} was derived using Equation 4.14:

$$\rho_{sh} = \frac{\eta_{sh}}{r_{max}} = \frac{\int_0^{r_{max}} g_{prox}(r)\ dr}{r_{max}}, \tag{4.14}$$

where η_{sh} is the integral of the proximal distribution function and r_{max} the position of the first minimum of $g_{prox}(r)$ that defines the outer boundary of the hydration shell.

A similar definition can be obtained for the density of bulk water:

$$\rho_0 = \frac{\eta_0}{r_{max}} = \frac{\int_0^{r_{max}} g_{o-o}(r)\ dr}{r_{max}}, \tag{4.15}$$

with η_0 the integral of the standard oxygen-oxygen radial distribution function of bulk water. Taking $r_{max} = 4.9$Å in Equations 4.14 and 4.15 allows comparison of the hydration shell density with that of the bulk: $\rho_{sh} = \rho_0(\eta_{sh}/\eta_0)$. Figures 4.5c-4.5f show the hydration shell density of lignin to be smaller than the bulk by 2%, 4%, 7% and 10% at $T = 300$K, 360K, 420K and 480K, respectively. This decrease in the hydration shell density is independent of geometric contributions that are also present if the hydration water is unperturbed from the bulk (Merzel and Smith, 2002).

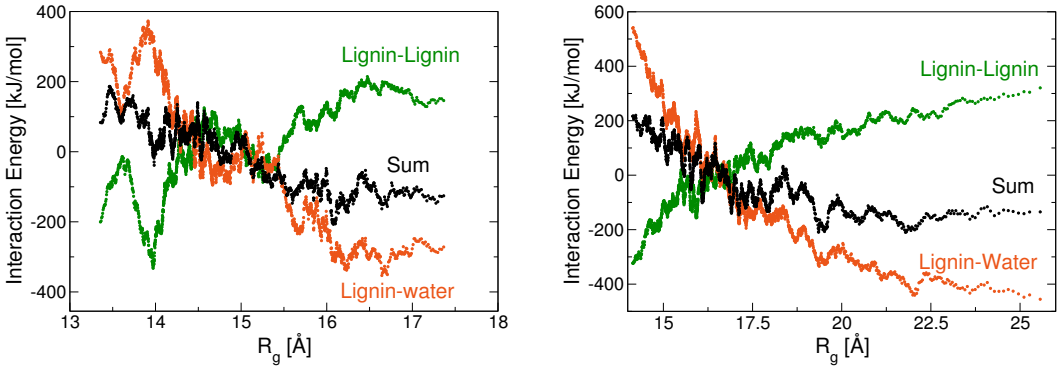
A hydration-shell water molecule participates on average in fewer hydrogen bonds than a bulk water molecule (Figure 4.5b), consistent with $\rho_{sh} < \rho_0$. Furthermore, the surface of lignin disrupts the hydrogen-bond network of its surrounding water. Although the area of the lignin:hydration-shell interface is only $\sim 10\%$ smaller than that of the bulk-water:hydration-shell "interface", hydration shell water molecules form 50% fewer hydrogen bonds with lignin than with the bulk.

4.3.5 Thermodynamics of the Collapse Transition

The thermodynamics of the transition of lignin from an extended conformation ($R_g = R_{ext}$, SASA= A_{ext}, $W = W_{ext}$) to a collapsed conformation ($R_g = R_{col} < R_{ext}$, SASA= A_{col}, $W = W_{col} < W_{ext}$) at 300K are now examined.

Enthalpy Change

For the compact lignin structures, $R_g \lesssim 14.2$Å, that are most frequently sampled in the 300K simulation, the intra-lignin interaction energy is negative (Figure 4.6a), but the lignin-water interaction energy is positive and of larger magnitude. In contrast, for extended lignin structures ($R_g \gtrsim 15.3$Å) that are rarely sampled, the lignin-water interaction energy is now negative and out-weighs the positive intra-lignin interaction,



(a) $T = 300$K **(b)** $T = 480$K

Figure 4.6: Lignin-lignin and lignin-water interactions energies as a function of the lignin R_g at (a) 300K; (b) 480K. Data represent ensemble average of the unbranched and one-branch lignins.

thus leading a net negative interaction energy. Thus, the collapse transition of lignin from $R_g = 16.3$Å to 14.1Å (Figure C.1) is enthalpically strongly disfavored, by $\Delta H \sim$ 200 kJ/mol.

Entropy of Hydration Water

Differences in water dynamics between the bulk and the hydration shell water molecules in the collapsed and extended lignin simulations can be analyzed by computing translational and rotational velocity autocorrelation functions (VACF) and the associated density of states, Figure 4.7. The translational VACFs of the hydration shell of extended and collapsed lignins are similar and both different to the bulk (Figure 4.7a). Negative values of the VACF are due to water molecules rebounding from collisions with their neighbors. The deeper negative minimum in the hydration shell arises from water confinement by the lignin surface, and has been also observed for hydration water of proteins (Abseher et al., 1996; Rocchi et al., 1998), DNA (Jana et al., 2006) and lipid bilayers (Debnath et al., 2010).

A peak in the translational density of states represents the population of a mode of a given frequency (Figure 4.7c). Therefore, the slight shift of the main peak of the hydration water towards higher frequencies arises from water molecules on the surface

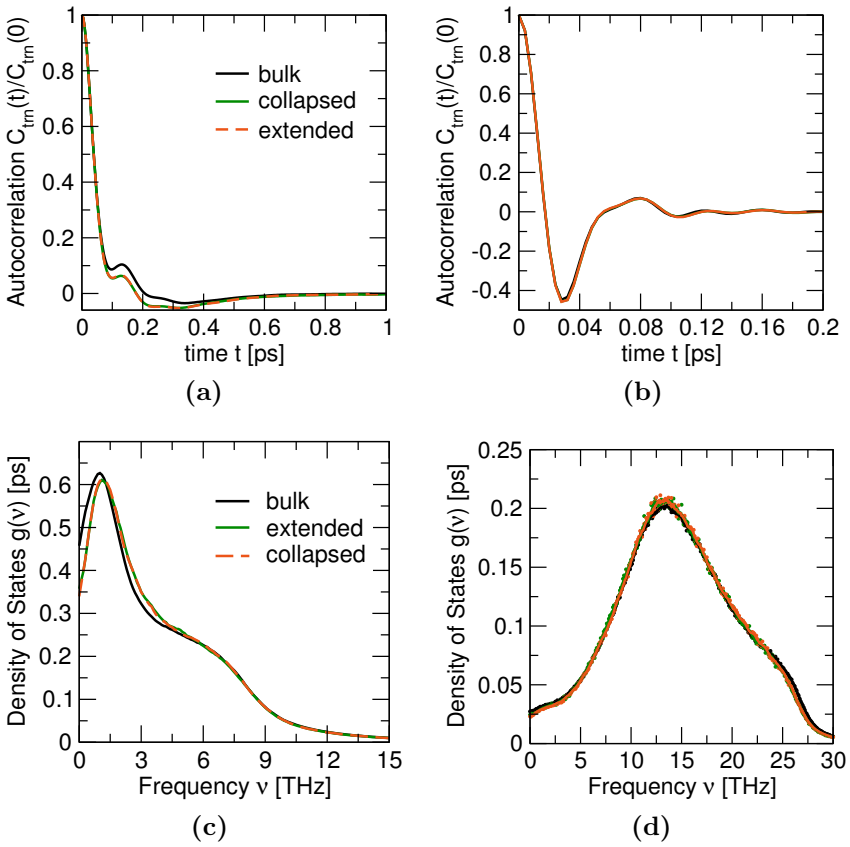


Figure 4.7: (a) Translational and (b) rotational velocity autocorrelation functions and the respective (c) translational and (d) rotational density of states of water.

Table 4.2: Comparison of entropy and fluidity of bulk and hydration water of collapsed and extended lignin structures at 300K

		bulk	*hydration collapsed*	*hydration extended*
trans. diffusion const. D_T	$[\text{Å}^2/ps]$	0.52±0.03	0.41±0.04	0.39±0.03
rotational diffusion const. D_R	$[1/ps]$	0.48±0.02	0.43±0.02	0.41±0.01
trans. fluidicity factor f_T		0.34±0.01	0.30±0.01	0.29±0.01
rotational fluidicity factor f_R		0.078±0.001	0.074±0.002	0.071±0.001
translational entropy TS_T	[kJ/mol]	16.32±0.06	15.79±0.03	15.76±0.08
rotational entropy TS_R	[kJ/mol]	3.952±0.008	3.951±0.009	3.925±0.006
total entropy TS	[kJ/mol]	20.28±0.06	19.74±0.04	19.69±0.09
entropy diff. $WT(S_{bulk} - S)$	[kJ/mol]	–	333±22	424±34

of the lignin that librate at a higher frequencies than the bulk (Debnath et al., 2010; Rocchi et al., 1998). Also, the translational diffusion coefficient,

$$D_T = \frac{k_B T g(\omega = 0)}{12 W m} \tag{4.16}$$

of hydration water is smaller than the bulk, supporting the idea that water molecules on the lignin surface are translationally restricted (see Table 4.2). The rotational VACF and $g(\omega)$ spectra show almost no variation between the bulk and hydration water, although the rotational diffusion coefficient is also higher for the bulk (see Table 4.2).

Consistent with the above, the translational and orientational entropies per water molecule, calculated using Equation 4.12, increase when going from the hydration water to the bulk (Table 4.2). Therefore, the hydration water molecules are entropically unfavorable compared to the bulk, by an associated free energy of $\sim$ 0.57 kJ/mol per molecule. However, no difference is found between the water hydrating extended and collapsed lignins. Consequently, when lignin collapses from the extended state, with W_{ext} water molecules in its solvation shell, to the collapsed state with $W_{col} < W_{ext}$ water molecules, then a number $(W_{ext} - W_{col})$ of hydration

waters are released to bulk, and the associated free energy change is thus given by:

$$\Delta G_{trns} = (W_{ext} - W_{col})\, T\, (S_{hydr} - S_{bulk})\,. \tag{4.17}$$

Thus, ΔG_{trns} favors lignin collapse by $\Delta G_{trns} = -91$ kJ/mol at $T = 300$K.

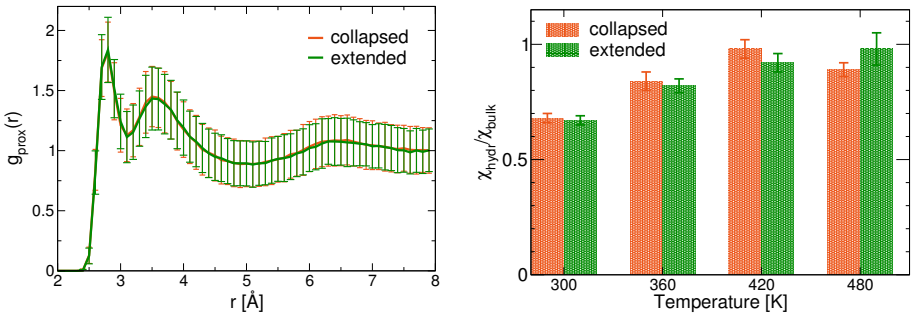
Hydration Water Density and Compressibility

The proximal distribution functions, and therefore the density, of the hydration water of extended and collapsed lignins are identical at 300K (Figure 4.8a). This demonstrates that the lignin collapse is not accompanied by dewetting. Similar behavior is found for "'small" hydrophobic solutes, as defined in the length-scale dependent theory of hydrophobicity (Chandler, 2005; Lum et al., 1999),where hydrogen bonds between water molecules solvating a hydrophobic solute remain intact. Therefore, the solvation free energy has an entropic origin and is given by the excess chemical potential, $\Delta\mu$ required to create a cavity of volume V in water of compressibility χ (Hummer et al., 1996; Garde et al., 1996):

$$\Delta\mu = \frac{V}{2\chi} + \frac{k_B T}{2} \log\left(2\pi\sigma_W^2\right), \tag{4.18}$$

where $\sigma_W = \langle W^2\rangle - \langle W\rangle^2$. $\Delta\mu$ in Equation 4.18 is roughly proportional to V (Chandler, 2005). Therefore, since the volumes of the extended and collapsed states are the same, the entropic penalty for creating a cavity to accommodate the collapsed conformation is equal to that for the extended state.

However, at 300K the compressibility of the lignin hydration water, χ_{hydr}, is lower than that of bulk water, χ_{bulk} (Figure 4.8b). The entropic cost of surrounding lignin with the less compressible hydration water equals the excess chemical potential of creating a cavity in bulk water(the volume of which equals that of the solvation shell)



(a) **(b)**

Figure 4.8: (a) Proximal distribution functions of water oxygen atoms at a distance r from the surface of the lignin at $T = 300$K. (b) Relative compressibility of hydration water. Data are derived from six 50ps MD trajectories: three with lignin in a collapsed state and three in an extended state.

and filling it with hydration water:

$$G_{fluc} = \frac{W}{2\rho}\left(\frac{1}{\chi_{hydr}} - \frac{1}{\chi_{bulk}}\right) + \frac{k_B T}{2}\log\left(\frac{\chi_{hydr}}{\chi_{bulk}}\right), \tag{4.19}$$

where the hydration shell has volume $V = W\rho$, density ρ, and contains W water molecules. Thus, the associated free energy change when lignin collapses from the extended state, the solvation shell of which has W_{ext} water molecules, to the collapsed state with $W_{col} < W_{ext}$ is given by:

$$\Delta G_{fluc} = \frac{W_{col} - W_{ext}}{2\rho}\left(\frac{1}{\chi_{hydr}} - \frac{1}{\chi_{bulk}}\right). \tag{4.20}$$

The ratio χ_{hydr}/χ_{bulk} is 0.82 ± 0.06, 0.82 ± 0.03, 0.95 ± 0.07 and 1.05 ± 0.21 at 300K, 360K, 420K and 480K, respectively. Thus, collective solvent density fluctuations favor the lignin collapse ($\Delta G_{fluc} < 0$) at $T \leq 360$, but do not influence the transition at high temperatures where $\chi_{hydr} \backsimeq \chi_{bulk}$. Substituting the parameters of Figures 4.8b and C.1 in Equation 4.20 gives at $\Delta G_{fluc} = -297$ kJ/mol at 300K.

Water compressibility has been recently identified as a measure of the hydrophobicity of molecular surfaces (Sarupria and Garde, 2009; Jamadagni et al., 2011; Godawat et al., 2009): with the more hydrophobic the surface the larger

the compressibility. Figure 4.8b would then imply that at 480K lignin is slightly more hydrophobic than at 300K. Interestingly, the fraction of hydrophilic SASA, $\langle\phi\rangle_{pol} = 0.42 \pm 0.01$, is the same for both the collapsed and extended conformations at all temperatures. This suggests that, although the polarity of the surface groups is the main factor determining hydrophobicity of a lignin polymer, the total SASA of the polymer, which is larger at high temperatures (Figure 4.4b), may play a secondary role.

Conformational Entropy of Lignin

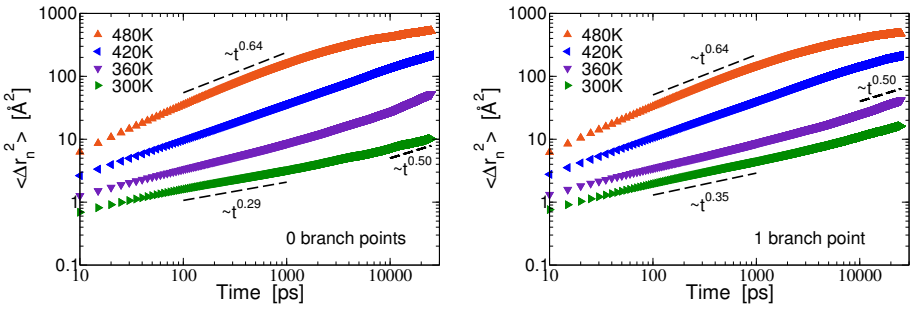
The extended lignin structures have larger conformational entropy, and the entropy change going from an extended (R_{ext}) to a collapsed (R_{col}) state can be estimated from the entropy penalty of confining a Gaussian polymer of R_{ext} to a volume $L \sim D^3$ of characteristic dimension $L = R_{col}$, $\Delta G_{conf} \simeq k_B T(\pi^2/3)\,(D/R_{ext})^2 = k_B T(\pi^2/3)\,(R_{col}/R_{ext})^2$ (Edwards and Freed, 1969). Using this method, here $\Delta G_{conf} = 11$ kJ/mol.

Free Energy Change of Lignin Collapse

Summing all the contributions discussed in detail above, the total free energy change of lignin collapse at 300K is approximately

$$\Delta G \simeq \Delta H + \Delta G_{trns} + \Delta G_{fluc} + \Delta G_{conf}. \tag{4.21}$$

The above calculations give $\Delta H \simeq 200$ kJ/mol, $\Delta G_{trns} \simeq -90$ kJ/mol, $\Delta G_{fluc} \simeq -300$ kJ/mol and $\Delta G_{conf} \simeq 10$ kJ/mol, and therefore the overall free energy of collapse is $\Delta G \simeq -180$ kJ/mol. Hence the release of entropically unfavorable hydration shell water molecules leading to ΔG_{trns} and ΔG_{fluc} drives the lignin collapse. This mechanism of collapse is different from that usually considered for coil to globule transitions, in which the favorable enthalpy gain arising from increased



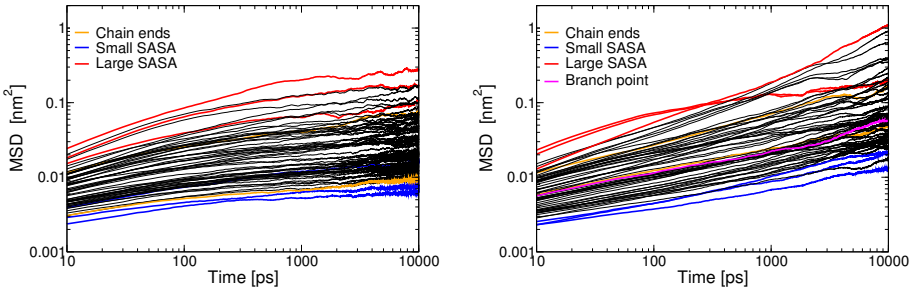
(a) L0 **(b)** L1

Figure 4.9: (a) Mean square displacements (MSD) of the ensemble of lignins with no branch points, $L0$, at four temperatures, with translation and rotation of the entire molecule removed. (b) MSD for the ensemble with one branch point, $L1$.

monomer:monomer contacts in the collapsed state compensates for the decrease in chain configurational entropy (Grosberg and Khokhlov, 1994).

4.3.6 Lignin Chain Dynamics

Time-dependent monomer mean-square displacements (MSD, Equation 4.5) from the ensemble of unbranched polymers are shown at four temperatures in Figure 4.9a. At low temperatures (T = 300K and 360K) Figure 4.9a exhibits three regimes, typical of glass-forming polymers (Dokholyan et al., 2002; Aichele et al., 2003): (i) ballistic ($t \lesssim 30$ps); (ii) a region (100ps $\lesssim t \lesssim$ 10ns, in which $\langle \Delta r_n^2 \rangle \sim t^{\beta}$ with $\beta_{300\text{K}} = 0.29$ and $\beta_{360\text{K}} = 0.42$, reflecting the temporary confinement of the monomers by their nearest neighbors in a "caging'" effect, (iii) sub-diffusive $t \gtrsim 10$ns where $\beta \simeq 0.5$, consistent with the Rouse theory of unbranched polymer melts (Doi and Edwards, 1983). With increasing temperature the caging effect decreases and at high temperatures (T = 420K and 480K) the caging plateau disappears (Dokholyan et al., 2002). The MSD exponent of $\beta_{420\text{K}} = 0.58$ and $\beta_{480\text{K}} = 0.64$ is similar to that found for unbranched polymer melts above their glass transition temperature ($\beta = 0.61$) (Paul and Smith, 2004). The increased chain mobility is also apparent in the graphical representations of Figure 4.2.



(a) L0$_b$ **(b)** L1$_a$

Figure 4.10: MSDs of individual monomers of a single trajectory of a lignin polymer at $T = 300$K with (a) no branch points $L0_b$ and (b) one branch point $L1_a$. Translation and rotation of the entire molecule have been removed. Monomers are highlighted with largest (red) and smallest SASA (blue) as well as monomers at chain ends (orange) and the branch point (magenta) .

The MSD of the one-branch-point ensemble, Figure 4.9b, displays characteristics similar to those for the zero branch polymers in Figure 4.9a. The power-law behavior exponents $< \Delta r_n^2 > \sim t^\beta$ are now $\beta_{300\text{K}} = 0.35$, $\beta_{360\text{K}} = 0.40$, $\beta_{420\text{K}} = 0.58$ and $\beta_{480\text{K}} = 0.64$. Overall, the MSD observed here at $T = 300$K is similar to that found in a previous MD study of lignin aggregates (Petridis et al., 2011).

Representative MSDs of individual monomers belonging to the same chain from a single MD simulation at $T = 300$K for an unbranched lignin molecule $L0_b$ and a one-branch molecule $L1_a$ are shown in Figures 4.10a and 4.10b, respectively. Significant variations are seen. Not all monomers are equally exposed to the solvent, leading to a range of solvent accessible surface areas (SASA) per monomer. A general trend is observed, in which monomers with smaller SASAs tend to have the lower $< \Delta r_n^2 >$. Interestingly, monomers at the ends of the chains do not display the fastest dynamics and nor do those at branch points exhibit the slowest dynamics. Similar behavior of $< \Delta r_n^2 >$ is observed for lignins with branch points (see Figure C.3 in appendix C).

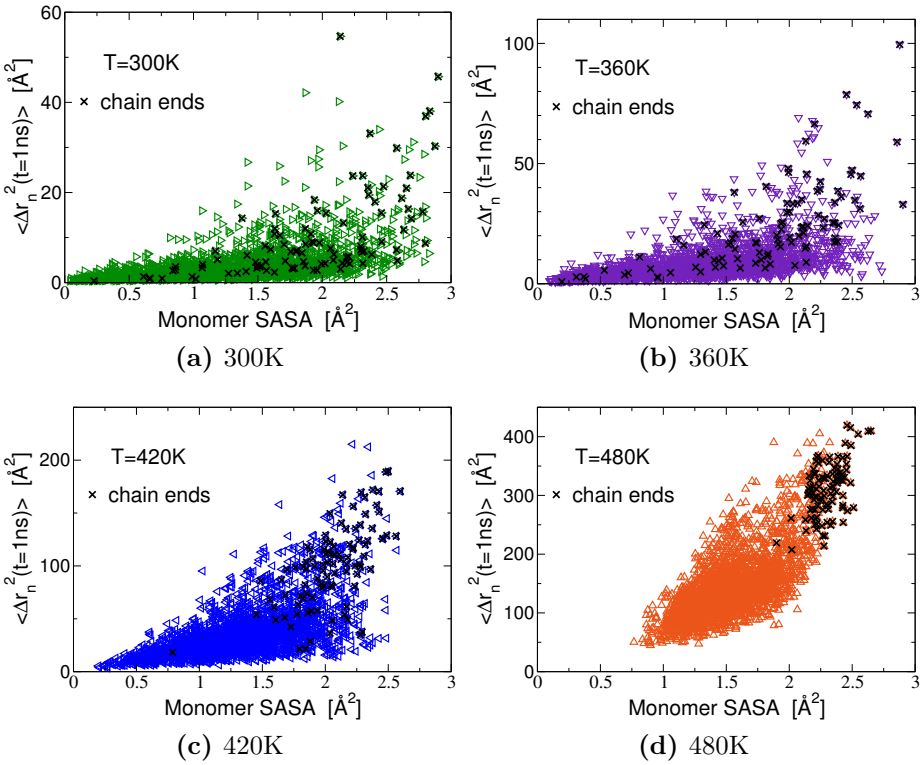
The dependence of the mobility of a monomer on its solvent exposure is stronger at high temperatures (Figure 4.11). The slopes of approximate linear regressions at $T = 300$K, 360K, 420K and 480K are 3, 6, 30 and 100 respectively. Furthermore, the strength of the correlation between monomer MSD and SASA

also increases with temperature, with approximate linear regression χ^2 coefficients in Figures 4.11a, 4.11b, 4.11c and 4.11d of 0.53, 0.55, 0.62 and 0.75, respectively.

4.4 Discussion

Pretreatment is responsible for a significant fraction of the production cost of cellulosic ethanol from lignocellulosic biomass, due to the associated energy requirements as well as capital and operating costs. Available information on pretreated lignin includes how much remains after pretreatment (Yang and Wyman, 2004; Wyman et al., 2005), its chemical composition (Samuel et al., 2010; Jung et al., 2010; Studer et al., 2011; Fu et al., 2011) and the size of the lignin aggregates that form after pretreatment (Petridis et al., 2011; Pingali et al., 2010; Donohoe et al., 2008; Chundawat et al., 2011). Commonly-employed experimental methods to examine lignin include analytical chemistry (Yang and Wyman, 2004; Wyman et al., 2005), NMR (Samuel et al., 2010), neutron scattering (Petridis et al., 2011; Pingali et al., 2010) and electron microscopy exploring the μm scale (Selig et al., 2007; Donohoe et al., 2008; Kristensen et al., 2008; Chundawat et al., 2011). The atomistic simulations presented here complement these studies by providing a detailed description of the temperature-dependent change in structure and dynamics of individual lignin molecules on the nm scale.

At low temperatures ($T \lesssim 420$K) lignins are found to be compact ellipsoidal objects, as indicated by the agreement between V_{SAS} and V_{gyr} (Figure 4.1), of low solvent accessible surface area. Contrary to what is found for polymers in good solvents, branching does not lead to reduction of the lignin R_g, and instead a strong correlation is found between the lignin R_g and its asphericity (Table 4.1), as one expects for compact ellipsoids. The polymers are comprised of inter-permeable $\sim$ 30-monomer blobs, with power-law chain statistics observed only inside the blobs (Figure 4.4). The compact lignin structures at $\lesssim$ 420K found here are consistent with small-angle neutron scattering studies showing lignin aggregation after dilute acid pretreatment at $\sim$ 120K (Pingali et al., 2010).



(a) 300K **(b)** 360K

(c) 420K **(d)** 480K

Figure 4.11: Monomer MSD at $t = 1$ns of the ensemble of polymers with zero and one branch points versus the monomer SASA.

Above 420K the lignin molecules expand to non-ellipsoidal, extended forms without cavities, leading to sharp increases in R_g, V_{gyr}, and A_{SAS} but no significant change in V_{SAS} or the asphericity Δ (Figure 4.1). The blobs become spatially separated, leading to self-similar chain packing, *i.e.* the same for segments of all lengths.

To characterize the thermodynamics of the lignin transition from extended to compact states at 300K, the various contributions to the free energy of collapse were investigated. The enthalpy change is positive (unfavorable), because the lignin:water interaction, favoring the extended state, is stronger than the lignin:lignin interaction that favors the collapse (Figure 4.6a). The lignin conformational entropy also does not favor the collapse, as extended states sample more configurational space. Hydration shell water molecules were found to have lower translational entropy than the bulk, while their rotational entropy was similar (Table 4.2). Additionally, hydration water has slightly lower density fluctuations than the bulk, making the latter entropically more favored. Protein unfolding, accompanied by exposure of core hydrophobic residues to water, has been recently associated with increase in hydration shell compressibility (Sarupria and Garde, 2009). Here the hydration water of collapsed and extended lignins have a similar compressibility, which, critically, is lower than that of the bulk.

The collapse of lignin from extended (large SASA) to compact (small SASA) states is accompanied by the displacement of hydration water molecules to the bulk. Therefore, the simulations demonstrate that the release of entropically unfavorable hydration water molecules into the bulk is the driving force behind the collapse of lignin at 300K.

Previous MD simulations of non-polar hydrophobic polymers in water found the hydration contribution to the free energy of collapse, defined as the total free energy minus the polymer:water interaction energy and intrapolymer interaction energy and entropy, to favor the collapse and to be enthalpy dominated (Athawale et al., 2007). The present lignin simulations are of larger polymers that Ref. 21, and, unlike

Ref.21, include water:polymer attractive Coulombic interactions. The results also show the hydration contribution (here approximated by $\Delta G_{trns} + \Delta G_{fluc}$) to drive the collapse, but to be entropy driven. The present simulations also complement previous calculations on the kinetics of hydrophobic collapse that found collective water density fluctuations as the rate limiting step in the collapse (Miller et al., 2007). Here, water density fluctuations are found to also contribute significantly to the thermodynamics of lignin collapse.

Although important biological functions of lignin, such as water conduction and cell wall defense against enzymatic digestion, stem directly from its hydrophobicity, the lignin surface is shown here to be wetted by water, as indicated by the lignin hydration water density being equal to the bulk (Figure 4.5c). Furthermore, the suppression of hydration water compressibility (Figure 4.8b) is similar to that found near hydrophilic surfaces.

Temperature-induced changes in lignin dynamics are highlighted by the disappearance of the intermediate-time (100ps − 10ns) plateau in the monomer mean-square displacements for temperatures above 360K. The low-temperature plateau, reflecting a temporary localization of the monomers by their nearest neighbors, is similar to that found for compact homopolymers that exhibit glassy behavior (Dokholyan et al., 2002). Interestingly, the onset of constrained dynamics below 360K occurs at lower temperature than the collapse transition at 420K. Monomers exposed to the solvent experience smaller friction than those buried in the core of the lignin, leading to a correlation between a monomer's SASA and its mobility (Figure 4.11). Therefore, the onset of the dynamic transition above 360K coincides with an increase with the overall lignin solvent exposure (Figure 4.1c). For polymer melts above their Θ-temperature (the temperature at which the second viral coefficient disappears and the polymer acts as an ideal Gaussian coil) chain connectivity determines monomer mobility, with chain ends exhibiting faster dynamics (Paul and Smith, 2004; Binder et al., 2003), something not always observed in Figures 4.11a and 4.11b.

4.5 Conclusions

Extensive, 17.5μs simulations of single lignin polymers in aqueous solution have probed the temperature-dependent structural and dynamic changes of this biomass component and its hydration water. With increasing temperature the lignin was found to transition from compact conformations with glassy dynamics to extended conformations with enhanced dynamics. The unfavorable translational entropy of lignin hydration water molecules and the low compressibility of the hydration shell were found to thermodynamically drive the transition from extended to collapsed states at 300K. The present molecular level understanding of the structure, dynamics and thermodynamics of lignin as a function of temperature may provide fundamental information needed to help understand biomass pretreatment and thus improve the efficiency of cellulosic ethanol production.

Chapter 5

Conclusions

Chapter 3 demonstrates that by using reaction field (RF), improving memory usage, and improving load balancing it is possible for the first time to simulate multi-million atom system accurately and efficiently with GROMACS. Chapter 2 and 3 with additional details in Appendix A and B, explain the new implementation of particle-mesh Ewald (PME) with 2D decomposition and OpenMP parallelization. This allows simulating extremely large systems even if they could not be modeled accurately with RF. The PME implementation scales to millions of atoms and tens of thousand of cores (additional results in Appendix D).

These methodological improvements allowed detailed simulations of large lignocellulose models to investigate the role of lignin in recalcitrance. The paper Lindner et al. (2013) presents the results of simulations performed with the system setup and validated as described in chapter 3. It suggests that the recalcitrance of crystalline cellulose to hydrolysis arises not only from the inaccessibility of inner fibers but also due to the promotion of lignin adhesion. An even larger model of post pretreated lignocellulose including enzymes was created and contained 23.67 million atoms. For its simulation further methodological improvements were required. I implemented virtual sites into the lignocellulose and glycosylation force-fields and validated the 4fs time-steps with virtual sites. The results are presented in Vermaas

et al. (2015). It shows that lignin binds exactly where for industrial purposes it is least desired, providing a simple explanation of why hydrolysis yields increase with lignin removal. Chapter 4 describes the study of the temperature dependent structural and dynamic properties of lignin by itself. It finds that the low-temperature collapse is thermodynamically driven by the increase of the translational entropy and density fluctuations of water molecules removed from the hydration shell.

The large improvements to the performance and scalability of GROMACS was aided by the usage of modern software engineering principles. Chapter 2 describes the efforts undertaken to improve programming productivity in GROMACS. This included a partial conversion of code from C language to the object-oriented C++ language. C++ is a very complex language and its complexity can be an additional entrance hurdle for domain scientists who want to contribute to the GROMACS code. To mitigate this problem, we developed with my contribution a coding standard to utilize a subset of C++ which improves productivity but limits the initial learning effort required. We introduced code review and continuous integration. In both cases I did the initial selection of best methods and implementation. Additionally we developed, as part of the Eclipse Parallel Tools Platform (PTP), a method to seamlessly use integrated development environments (IDE) remotely. Application development for supercomputers always requires remote development and thus this capability of an IDE is of particular importance. The new method is called synchronized remote projects and was designed and co-developed by me. The trend of increased number of compute cores will continue. Our porting and SIMD improvements for Xeon Phi are described in chapter 2. Additionally I made it possible to use more than 32 OpenMP threads and improved the OpenMP scaling by adding OpenMP parallelization to loops which only become a bottleneck at large number of threads.

Exascale machines will require up to a billion parallel threads. As described in chapter 2, GROMACS has already implemented three levels of parallelism: SIMD/SIMT, OpenMP/thread, and MPI/domain-decomposition. To improve scaling

to exascale, extra levels of parallelization will be required. All MD force calculations are independent and can be done in parallel to each other. While this is being used by the GPU or Phi offloading, it is currently not exploited within a single CPU. In the future task-parallelism will allow us to exploit this and can be implemented with the threading building blocks library. In addition, enhanced sampling methods such as Replica-Exchange and Markov-State modeling can be used to combine many independent MD simulations. While these methods are available now, they are not yet routinely used because of problems related to scalability to larger systems, efficiency, and usability. Improvements to these methods are needed to apply them efficiently to larger systems. Better integration into GROMACS will improve the performance and usability. Together this will allow MD simulations to apply at least five level of parallelism: the existing three levels and additionally both task-level and enhanced sampling. The existing parallelism will need to be further optimized, in particular the electrostatic method. The fast-multipole and multigrid methods allow calculation of the Coulomb interaction as accurately as with PME, without requiring the all-to-all communication pattern. Further improvements to these methods are needed to make them as computationally efficient. All optimized five levels combined will be needed to provide sufficient parallelism at exascale.

To achieve the full potential, future MD simulations will need to increase the accessible time-scale and not just length-scale. Longer simulations are needed to be able to compare results to experiments with longer intrinsic observation time. They are also essential to be able to make chemical, biological and physiological relevant predictions for any process with longer time-scale. Most simulations of larger biological systems will require longer simulations because most processes on these scales happen on longer time-scale. Reaching longer time-scales will require a single time-step to be computed in ever shorter time. Given that the extra computing performance will originate from extra parallelism, improvements of the time-scale is only possible with improved strong scaling. This will be challenging because fewer and fewer atoms will be computed by a single compute unit. And

it is hard to imagine that strong scaling is possible to less than a single atom. Two hardware trends have the potential to mitigate the problem. The continuing integration of computer components will lead to the integration of computing and optical networking. Besides improving power efficiency, this has the potential to decrease the latency of communicating between nodes and thus decrease the limit network latency places on the minimal time-step. To reach even shorter time-steps, the entire MD simulation will have to be computed on a single node. The continuing increase of performance of CPUs and GPUs will make this increasingly possible; initially only for small systems but eventually also for larger biological models. The communication between nodes will then only be used for enhanced sampling methods.

Bibliography

Abseher, R., Schreiber, H., and Steinhauser, O. (1996). The influence of a protein on water dynamics in its vicinity investigated by molecular dynamics simulation. *Proteins Struct. Funct. Bioinf.*, 25(3):366–378. 72

Aichele, M., Gebremichael, Y., Starr, F. W., Baschnagel, J., and Glotzer, S. C. (2003). Polymer-specific effects of bulk relaxation and stringlike correlated motion in the dynamics of a supercooled polymer melt. *J. Chem. Phys.*, 119(10):5290–5304. 78

Amadei, A., Linssen, A. B. M., and Berendsen, H. J. C. (1993). Essential dynamics of proteins. *PROTEINS: Struct. Funct. Gen.*, 17(4):412–425. 17, 133

Ashbaugh, H. S., Pratt, L. R., Paulaitis, M. E., Clohecy, J., and Beck, T. L. (2005). Deblurred observation of the molecular structure of an oil-water interface. *J. Am. Chem. Soc.*, 127(9):2808–2809. 58

Athawale, M. V., Goel, G., Ghosh, T., Truskett, T. M., and Garde, S. (2007). Effects of lengthscales and attractions on the collapse of hydrophobic polymers in water. *PNAS*, 104(3):733–738. 52, 82

Becker, O., MacKerell, A., Roux, B., and Watanabe, M. (2001). In *Computational Biochemistry and Biophysics*. Marcel-Decker, Inc., 1st ed edition. 1, 24

Bekker, H., Berendsen, H. J. C., Dijkstra, E. J., Achterop, S., v. Drunen, R., v. d. Spoel, D., Sijbers, A., Keegstra, H., Reitsma, B., and Renardus, M. K. R. (1993a). Gromacs Method of Virial Calculation Using a Single Sum. In de Groot, R. A. and Nadrchal, J., editors, *Physics Computing 92*, pages 257–261, Singapore. World Scientific. 13

Bekker, H., Berendsen, H. J. C., Dijkstra, E. J., Achterop, S., van Drunen, R., van der Spoel, D., Sijbers, A., Keegstra, H., Reitsma, B., and Renardus, M. K. R. (1993b). Gromacs: A parallel computer for molecular dynamics simulations. In de Groot, R. A. and Nadrchal, J., editors, *Physics Computing 92*, pages 252–256, Singapore. World Scientific. 9

Berendsen, H. J., van der Spoel, D., and van Drunen, R. (1995a). Gromacs - a message-passing parallel molecular-dynamics implementation. *Comput. Phys. Commun.*, 91(1-3):43–56. 9

Berendsen, H. J. C., Postma, J. P. M., van Gunsteren, W. F., Dinola, A., and Haak, J. R. (1984). Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81(8):3684–3690. 30

Berendsen, H. J. C., van der Spoel, D., and van Drunen, R. (1995b). GROMACS: A message-passing parallel molecular dynamics implementation. *Comp. Phys. Comm.*, 91:43–56. 55

Berendsen, H. J. C. and van Gunsteren, W. F. (1984). Molecular dynamics simulations: Techniques and approaches. In et al., A. J. B., editor, *Molecular Liquids-Dynamics and Interactions*, NATO ASI C 135, pages 475–500. Reidel, Dordrecht, The Netherlands. 16

Berne, B. J., Weeks, J. D., and Zhou, R. H. (2009). Dewetting and hydrophobic interaction in physical and biological systems. *Annu. Rev. Phys. Chem.*, 60:85–103. 52

Besombes, S. and Mazeau, K. (2005). The cellulose/lignin assembly assessed by molecular modeling. part 2: seeking for evidence of organization of lignin molecules at the interface with cellulose. *Plant Physiology and Biochemistry*, 43(3):277–286. 53

Binder, K., Baschnagel, J., and Paul, W. (2003). Glass transition of polymer melts: test of theoretical concepts by computer simulation. *Prog. Polym. Sci.*, 28(1):115–172. 83

Bjurhager, I., Olsson, A.-M., Zhang, B., Gerber, L., Kumar, M., Berglund, L. A., Burgert, I., Sundberg, B., and Salmen, L. (2010). Ultrastructure and mechanical

properties of populus wood with reduced lignin content caused by transgenic down-regulation of cinnamate 4-hydroxylase. *Biomacromol.*, 11(9):2359–2365. 52

Bowers, K. J., Chow, E., Xu, H., Dror, R. O., Eastwood, M. P., Gregersen, B. A., Klepeis, J. L., Kolossvary, I., Moraes, M. A., Sacerdoti, F. D., Salmon, J. K., Shan, Y., and Shaw, D. E. (2006). Scalable algorithms for molecular dynamics simulations on commodity clusters. In *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, New York, NY, USA. ACM. 7, 24, 47

Bowers, K. J., Dror, R. O., and Shaw, D. E. (2005). Overview of neutral territory methods for the parallel evaluation of pairwise particle interactions. *J. Physics: Conference Series*, 16(1):300. 9

Bowers, K. J., Dror, R. O., and Shaw, D. E. (2007). Zonal methods for the parallel execution of range-limited n-body simulations. *J. Comput. Phys.*, 221(1):303–329. 9

Brooks, B. R., Bruccoleri, R. E., Olafson, B. D., States, D. J., Swaminathan, S., and Karplus, M. (1983). CHARMM - a program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.*, 4(2):187–217. 7, 31, 132

Brunow, G., Kilpelainen, I., Lapierre, C., Lundquist, K., Simola, L. K., and Lemmetyinen, J. (1993). The chemical structure of extracellular lignin released by cultures of picea abies. *Phytochemistry*, 32(4):845–850. 54

Bulacu, M., Goga, N., Zhao, W., Rossi, G., Monticelli, L., Periole, X., Tieleman, D., and Marrink, S. (2005). Improved angle potentials for coarse-grained molecular dynamics simulations. *J. Chem. Phys.*, 123(11):3282–92. 18

Bussi, G., Donadio, D., and Parrinello, M. (2007). Canonical sampling through velocity rescaling. *J. Chem. Phys.*, 126(1):014101. 20, 55

Caleman, C., van Maaren, P. J., Hong, M., Hub, J. S., Costa, L. T., and van der Spoel, D. (2012). Force field benchmark of organic liquids: Density, enthalpy of vaporization, heat capacities, surface tension, isothermal compressibility, volumetric expansion coefficient, and dielectric constant. *J. Chem. Theory Comput.*, 8(1):61–74. 19

Case, D. A., Cheatham, T. E., Darden, T., Gohlke, H., Luo, R., Merz, K. M., Onufriev, A., Simmerling, C., Wang, B., and Woods, R. J. (2005). The Amber biomolecular simulation programs. *J. Comput. Chem.*, 26(16):1668–1688. 7, 14

Chandler, D. (2005). Interfaces and the driving force of hydrophobic assembly. *Nature*, 437(7059):640–647. 52, 75

Chundawat, S. P. S., Donohoe, B. S., da Costa Sousa, L., Elder, T., Agarwal, U. P., Lu, F., Ralph, J., Himmel, M. E., Balan, V., and Dale, B. E. (2011). Multi-scale visualization and characterization of lignocellulosic plant cell wall deconstruction during thermochemical pretreatment. *Energy Environ. Sci.*, 4:973–984. 53, 80

Cosgrove, D. J. (2005). Growth of the plant cell wall. *Nat. Rev. Mol. Cell Biol.*, 6(11):850–861. 26

Dadarlat, V. M. and Post, C. B. (2001). Insights into protein compressibility from molecular dynamics simulations. *J. Phys. Chem. B*, 105(3):715–724. 58

Darden, T., York, D., and Pedersen, L. (1993). Particle mesh Ewald: An $N \cdot \log(N)$ method for Ewald sums in large systems. *J. Chem. Phys.*, 98(12):10089–10092. 2, 9, 25, 55

Debnath, A., Mukherjee, B., Ayappa, K. G., Maiti, P. K., and Lin, S.-T. (2010). Entropy and dynamics of water in hydration layers of a bilayer. *J. Chem. Phys.*, 133(17):174704. 53, 72, 74

DeGennes, P. (1975). Collapse of a polymer chain in poor solvents. *J. Phys. Lett.*, 36:55–57. 52

Ding, S. Y. and Himmel, M. E. (2006). The maize primary cell wall microfibril: A new model derived from direct visualization. *J. Agric. Food Chem.*, 54(3):597–606. 28

Doi, M. and Edwards, S. F. (1983). *Theory of polymer dynamics*. OUP, Oxford. 78

Dokholyan, N. V., Pitard, E., Buldyrev, S. V., and Stanley, H. E. (2002). Glassy behavior of a homopolymer from molecular dynamics simulations. *Phy. Rev. E*, 65(3):030801. 78, 83

Donohoe, B. S., Decker, S. R., Tucker, M. P., Himmel, M. E., and Vinzant, T. B. (2008). Visualizing lignin coalescence and migration through maize cell walls following thermochemical pretreatment. *Biotechnol. Bioeng.*, 101(5):913–925. 53, 80

Edwards, S. F. and Freed, K. F. (1969). Entropy of a collapsed polymer i. *J. Phys. A: Gen. Phys.*, 2(2):145–150. 77

Eleftheriou, M., Fitch, B., Rayshubskiy, A., Ward, T., and Germain, R. (2005). Performance measurements of the 3D FFT on the Blue Gene/L supercomputer. *Lecture notes in computer science*, 3648:795. 131

Essmann, U., Perera, L., Berkowitz, M. L., Darden, T., Lee, H., and Pedersen, L. G. (1995). A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103(19):8577–8593. 2, 25, 55

Fitch, B. G., Rayshubskiy, A., Eleftheriou, M., Ward, C. T. J., Giampapa, M., Pitman, M. C., and Germain, R. S. (2006). Blue matter: approaching the limits of concurrency for classical molecular dynamics. In *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, New York, NY, USA. ACM. 47

Flory, P. (1966). *Principles of polymer chemistry*. Cornell University Press, Ithaca, N.Y. 52

Freddolino, P. L., Arkhipov, A. S., Larson, S. B., Mcpherson, A., and Schulten, K. (2006). Molecular dynamics simulations of the complete satellite tobacco mosaic virus. *Structure*, 14(3):437–449. 24

Freddolino, P. L., Liu, F., Gruebele, M. H., and Schulten, K. (2008). Ten-microsecond MD simulation of a fast-folding WW domain. *Biophys. J.*, 94(10):L75–L77. 47

Freire, J. J. (1999). Conformational properties of branched polymers: Theory and simulations. In *Branched Polymers II*, volume 143 of *Advances in Polymer Science*, pages 35–112. Springer-Verlag Berlin, Berlin. Review. 66

French, A. D. and Johnson, G. P. (2004). Advanced conformational energy surfaces for cellobiose. *Cellulose*, 11(3-4):449–462. 39, 134

Frigo, M. and Johnson, S. (2005). The design and implementation of FFTW 3. 93(2):216–231. 45, 131

Fritsch, S., Junghans, C., and Kremer, K. (2012). Structure formation of toluene around C60: Implementation of the adaptive resolution scheme (AdResS) into GROMACSgroma. *J. Chem. Theory Comput.*, 8:398–403. 18

Fu, C., Mielenz, J. R., Xiao, X., Ge, Y., Hamilton, C. Y., Rodriguez, M., Chen, F., Foston, M., Ragauskas, A., Bouton, J., Dixon, R. A., and Wang, Z.-Y. (2011). Genetic manipulation of lignin reduces recalcitrance and improves ethanol production from switchgrass. *PNAS*, 108(9):3803–3808. 80

Garde, S., Hummer, G., Garcia, A. E., Paulaitis, M. E., and Pratt, L. R. (1996). Origin of entropy convergence in hydrophobic hydration and protein folding. *Phys.Rev. Lett.*, 77(24):4966–4968. 52, 75

Gargallo, R., Hünenberger, P. H., Avilés, F. X., and Oliva, B. (2003). Molecular dynamics simulation of highly charged proteins: comparison of the particle-particle particle-mesh and reaction field methods for the calculation of electrostatic

interactions. *Protein science : a publication of the Protein Society*, 12(10):2161–2172. 27, 47, 48

Gennes, P. G. d. (1979). *Scaling concepts in polymer physics*. Cornell University Press, Ithaca, N.Y. 64

Godawat, R., Jamadagni, S. N., and Garde, S. (2009). Characterizing hydrophobicity of interfaces by using cavity formation, solute binding, and water correlations. *PNAS*, 106(36):1511oc6y6trb 9–15124. 76

Grabber, J. H. (2005). How do lignin composition, structure, and cross-linking affect degradability? a review of cell wall model studies. *Crop Sci.*, 45(3):820–831. 51

Grosberg, A. and Khokhlov, A. (1994). *Statistical physics of macromolecules*. AIP series in polymers and complex materials. AIP Press. 78

Grosberg, A., Rabin, Y., Havlin, S., and Neer, A. (1993). Crumpled globule model of the3-dimensional structure of dna. *Europhys. Lett.*, 23(5):373–378. 64

Grosberg, A. Y., Nechaev, S. K., and Shakhnovich, E. I. (1988). The role of topological contraints in the kinetics of collapse of macromolecules. *J. Phy.*, 49(12):2095–2100. 64

Gullingsrud, J. (2009). catdcd. http://www.ks.uiuc.edu/Development/MDTools/catdcd/. 31

Gullingsrud, J., Saam, J., and Phillips, J. (2006). psfgen. http://www.ks.uiuc.edu/Research/vmd/plugins/psfgen/. 31, 132

Gunsteren, W. F., Berendsen, H. J., and Rullmann, J. A. (1978). Inclusion of reaction fields in molecular dynamics. application to liquid water. *Faraday Discuss. Chem. Soc.*, 66:58–70. 26

Hansmann, U. H. and Okamoto, Y. (1997). Numerical comparisons of three recently proposed algorithms in the protein folding problem. *J. Comput. Chem.*, 18(7):920–33. 17

Henrion, U., Renhorn, J., Börjesson, S. I., Nelson, E. M., Schwaiger, C. S., Bjelkmar, P., Wallner, B., Lindahl, E., and Elinder, F. (2012). Tracking a complete voltage-sensor cycle with metal-ion bridges. *Proceedings of the National Academy of Sciences*, 109(22):8552–8557. 19

Hess, B. (2008). P-LINCS: A parallel linear constraint solver for molecular simulation. *J. Chem. Theory Comput.*, 4(1):116–122. 16, 19, 30, 55

Hess, B., Bekker, H., Berendsen, H. J., and Fraaije, J. G. (1997). LINCS: A linear constraint solver for molecular simulations. *J. Comput. Chem.*, 18(12):1463–1472. 16

Hess, B., Kutzner, C., van der Spoel, D., and Lindahl, E. (2008). GROMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation. *J. Chem. Theory Comput.*, 4(3):435–447. 9, 24, 27, 47, 48, 55, 131

Himmel, M. E., Ding, S. Y., Johnson, D. K., Adney, W. S., Nimlos, M. R., Brady, J. W., and Foust, T. D. (2007). Biomass recalcitrance: Engineering plants and enzymes for biofuels production. *Science*, 315(5813):804–807. 26, 51

Hoover, W. G. (1985). Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A*, 31(3):1695+. 30

Hummer, G., Garde, S., Garcia, A. E., Pohorille, A., and Pratt, L. R. (1996). An information theory model of hydrophobic interactions. *PNAS*, 93(17):8951–8955. 52, 75

Humphrey, W., Dalke, A., and Schulten, K. (1996). VMD: Visual molecular dynamics. *J. Mol. Graph.*, 14(1):33–38. 17

Irvine, G. M. (1985). The significance of the glass transition of lignin in thermomechanical pulping. *Wood Sci. Technol.*, 19(2):139–149. 52

Islam, M. F., Jenkins, R. D., Bassett, D. R., Lau, W., and Ou-Yang, H. D. (2000). Single chain characterization of hydrophobically modified polyelectrolytes using cyclodextrin/hydrophobe complexes. *Macromol.*, 33(7):2480–2485. 53

Jamadagni, S. N., Godawat, R., and Garde, S. (2011). Hydrophobicity of proteins and interfaces: Insights from density fluctuations. *Annu. Rev. Chem. Biomol. Eng.*, 2(1):147–171. 76

Jana, B., Pal, S., Maiti, P. K., Lin, S. T., Hynes, J. T., and Bagchi, B. (2006). Entropy of water in the hydration layer of major and minor grooves of dna. *J. Phys. Chem. B*, 110(39):19611–19618. 53, 72

Jorgensen, H., Kristensen, J. B., and Felby, C. (2007). Enzymatic conversion of lignocellulose into fermentable sugars: challenges and opportunities. *Biofuels, Bioprod. Biorefin.*, 1(2):119–134. 51

Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W., and Klein, M. L. (1983). Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79:926–935. 28, 55

Jung, S., Foston, M., Sullards, M. C., and Ragauskas, A. J. (2010). Surface characterization of dilute acid pretreated populus deltoides by tof-sims. *Energy Fuels*, 24:1347–1357. 80

Kasson, P. M., Lindahl, E., and Pande, V. S. (2010). Atomic-resolution simulations predict a transition state for vesicle fusion defined by contact of a few lipid tails. *PLoS Comput. Biol.*, 6(6). 125

Kristensen, J. B., Thygesen, L. G., Felby, C., Jorgensen, H., and Elder, T. (2008). Cell-wall structural changes in wheat straw pretreated for bioethanol production. *Biotechnol. Biofuels*, 1. 53, 80

Kubota, K., Fujishige, S., and Ando, I. (1990). Single-chain transition of poly(n-isopropylacrymide) in water. *J. Phys. Chem.*, 94(12):5154–5158. 52

Kuttel, M., Brady, J. W., and Naidoo, K. J. (2002). Carbohydrate solution simulations: Producing a force field with experimentally consistent primary alcohol rotational frequencies and populations. *J. Comput. Chem.*, 23(13):1236–1243. 27, 42

Kutzner, C., Czub, J., and Grubmüller, H. (2011a). Keep it flexible: Driving macromolecular rotary motions in atomistic simulations with GROMACS. *J. Chem. Theory Comput.*, 7:1381–1393. 18

Kutzner, C., Grubmüller, H., de Groot, B. L., and Zachariae, U. (2011b). Computational electrophysiology: the molecular dynamics of ion channel permeation and selectivity in atomistic detail. *Biophys. J.*, 101:809–817. 18

Lamb, H. (1920). *Higher Mechanics*. The University Press. 57

Larsson, D. S., Liljas, L., and van der Spoel, D. (2012). Virus capsid dissolution studied by microsecond molecular dynamics simulations. *PLoS Comput. Biol*, 8(5):e1002502. 125

Larsson, P. and Lindahl, E. (2010). A High-Performance Parallel-Generalized Born Implementation Enabled by Tabulated Interaction Rescaling. *J. Comp. Chem.*, 31(14):2593–2600. 118

Levitt, M. and Sharon, R. (1988). Accurate simulation of protein dynamics in solution. *PNAS*, 85(20):7557–7561. 70

Li, I. T. S. and Walker, G. C. (2010). Interfacial free energy governs single polystyrene chain collapse in water and aqueous solutions. *J. Am. Chem. Soc.*, 132(18):6530–6540. 52

Lieberman-Aiden, E., van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B. R., Sabo, P. J., Dorschner, M. O., Sandstrom, R., Bernstein, B., Bender, M. A., Groudine, M., Gnirke, A., Stamatoyannopoulos, J., Mirny, L. A., Lander, E. S., and Dekker, J. (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–293. 66

Lin, S. T., Blanco, M., and Goddard, W. A. (2003). The two-phase model for calculating thermodynamic properties of liquids from molecular dynamics: Validation for the phase diagram of lennard-jones fluids. *J. Chem. Phys.*, 119(22):11792–11805. 59

Lin, S. T., Maiti, P. K., and Goddard, W. A. (2005). Dynamics and thermodynamics of water in pamam dendrimers at subnanosecond time scales. *J. Phys. Chem. B*, 109(18):8663–8672. 53

Lin, S. T., Maiti, P. K., and Goddard, W. A. (2010). Two-phase thermodynamic model for efficient and accurate absolute entropy of water from molecular dynamics simulations. *J. Phys. Chem. B*, 114(24):8191–8198. 59

Lindahl, E., Hess, B., and van der Spoel, D. (2001). GROMACS 3.0: a package for molecular simulation and trajectory analysis. *J. Mol. Mod.*, 7(8):306–317. 9, 31, 55

Lindner, B., Petridis, L., Schulz, R., and Smith, J. C. (2013). Solvent-driven preferential association of lignin with regions of crystalline cellulose in molecular dynamics simulation. *Biomacromolecules*, 14(10):3390–3398. 85

Lum, K., Chandler, D., and Weeks, J. D. (1999). Hydrophobicity at small and large length scales. *J. Phys. Chem. B*, 103(22):4570–4577. 52, 75

Lundborg, M., Apostolov, R., Spångberg, D., Gärdenäs, A., van der Spoel, D., and Lindahl, E. (2014). An efficient and extensible format, library, and api for binary trajectory data from molecular simulations. *J. Comp. Chem.*, 35(3):260–269. 18

Lyubartsev, A. P., Martsinovski, A. A., Shevkunov, S. V., and Vorontsov-Velyaminov, P. N. (1992). New approach to Monte Carlo calculation of the free energy: Method of expanded ensembles. *J. Chem. Phys.*, 96:1776–1783. 18

Ma, J. P., Straub, J. E., and Shakhnovich, E. I. (1995). Simulation study of the collapse of linear and ring homopolymers. *J Chem. Phys.*, 103(7):2615–2624. 52, 64

Mackerell, A. (2004). Empirical force fields for biological macromolecules: Overview and issues. *J. Comput. Chem.*, 25(13):1584–1604. 25

Mackerell, A. D., Bashford, D., Bellott, M., Dunbrack, R. L., Evanseck, J. D., Field, M. J., Fischer, S., Gao, J., Guo, H., Ha, S., Joseph-McCarthy, D., Kuchnir, L., Kuczera, K., Lau, F. T., Mattos, C., Michnick, S., Ngo, T., Nguyen, D. T., Prodhom, B., Reiher, W. E., Roux, B., Schlenkrich, M., Smith, J. C., Stote, R., Straub, J., Watanabe, M., Wiorkiewicz-Kuczera, J., Yin, D., and Karplus, M. (1998). All-atom empirical potential for molecular modeling and dynamics studies of proteins. *J. Phys. Chem. B*, 102(18):3586–3616. 27

Malde, A. K., Zuo, L., Breeze, M., Stroet, M., Poger, D., Nair, P. C., Oostenbrink, C., and Mark, A. E. (2011). An automated force field topology builder (ATB) and repository: Version 1.0. *J. Chem. Theory Comput.*, 7(12):4026–4037. 19

Marrink, S. J., Risselada, H. J., Yefimov, S., Tieleman, D. P., and de Vries, A. H. (2007). The martini force field: Coarse grained model for biomolecular simulations. *J. Phys. Chem. B*, 111(27):7812–7824. 19

Mathias, G., Egwolf, B., Nonella, M., and Tavan, P. (2003). A fast multipole method combined with a reaction field for long-range electrostatics in molecular dynamics

simulations: The effects of truncation on the properties of water. *J. Chem. Phys.*, 118(24):10847–10860. 27, 35, 37, 47

Matthews, J. F., Skopec, C. E., Mason, P. E., Zuccato, P., Torget, R. W., Sugiyama, J., Himmel, M. E., and Brady, J. W. (2006). Computer simulation studies of microcrystalline cellulose i beta. *Carbohydr. Res.*, 341(1):138–152. 39, 42

McCammon, J. A., Gelin, B. R., and Karplus, M. (1977). Dynamics of folded proteins. *Nature*, 267(5612):585–590. 1, 24

McLain, S., Soper, A., Daidone, I., Smith, J., and Watts, A. (2008). Charge-based interactions between peptides observed as the dominant force for association in aqueous solution13. *Angewandte Chemie International Edition*, 47(47):9059–9062. 31

Merzel, F. and Smith, J. C. (2002). Is the first hydration shell of lysozyme of higher density than bulk water? *PNAS*, 99(8):5378–5383. 53, 58, 71

Miller, T. F., Vanden-Eijnden, E., and Chandler, D. (2007). Solvent coarse-graining and the string method applied to the hydrophobic collapse of a hydrated chain. *PNAS*, 104(37):14559–14564. 52, 83

Mitsutake, A., Sugita, Y., and Okamoto, Y. (2001). Generalized-ensemble algorithms for molecular simulations of biopolymers. *Biopolymers*, 60(2):96–123. 17

Miyamoto, S. and Kollman, P. A. (1992). SETTLE: An analytical version of the SHAKE and RATTLE algorithms for rigid water models. *J. Comp. Chem.*, 13:952–962. 55

Murail, S., Wallner, B., Trudell, J. R., Bertaccini, E., and Lindahl, E. (2011). Microsecond simulations indicate that ethanol binds between subunits and could stabilize an open-state model of a glycine receptor. *Biophysical journal*, 100(7):1642–1650. 125

Nina, M. and Simonson, T. (2002). Molecular dynamics of the trna(ala) acceptor stem: Comparison between continuum reaction field and particle-mesh ewald electrostatic treatments. *J. Phys. Chem. B*, 106(14):3696–3705. 27, 47, 48

Nishiyama, Y., Langan, P., and Chanzy, H. (2002). Crystal structure and hydrogen-bonding system in cellulose 1 beta from synchrotron x-ray and neutron fiber diffraction. *J. Am. Chem. Soc.*, 124(31):9074–9082. 28, 42

Oster, G. and Kirkwood, J. G. (1943). The influence of hindered molecular rotation on the dielectric constants of water, alcohols, and other polar liquids. *J. Chem. Phys.*, 11(4):175–178. 33

Páll, S., Abraham, M. J., Kutzner, C., Hess, B., and Lindahl, E. (2015). Tackling exascale software challenges in molecular dynamics simulations with GROMACS. In Markidis, S. and Laure, E., editors, *Solving Software Challenges for Exascale*, Lecture Notes in Computer Science, pages 3–27. Springer International Publishing. 9

Páll, S. and Hess, B. (2013). A flexible algorithm for calculating pair interactions on SIMD architectures. *Comp. Phys. Comm.*, 184(12):2641–2650. 10

Pan, A. C. and Roux, B. (2008). Building markov state models along pathways to determine free energies and rates of transitions. *J. Chem. Phys.*, 129(6):064107. 17

Parrinello, M. and Rahman, A. (1981). Polymorphic transitions in single crystals: A new molecular dynamics method. *J. Appl. Phys.*, 52(12):7182–7190. 30, 55

Paul, W. and Smith, G. D. (2004). Structure and dynamics of amorphous polymers: computer simulations compared to experiment and theory. *Rep. Prog. Phys.*, 67(7):1117–1185. 78, 83

Petridis, L., Pingali, S. V., Urban, V., Heller, W., O'Neill, H., Foston, M., Ragauskas, A., and Smith, J. C. (2011). Self-similar multiscale strucutre of lignin revealed by

neutron scattering and molecualr dynamics simulation molecular. *Phys. Rev. E*, 83:061911. 53, 54, 70, 79, 80

Petridis, L. and Smith, J. C. (2009). A molecular mechanics force field for lignin. *J. Comp. Chem.*, 30(3):457–467. 27, 55

Petridis, L., Xu, J., Crowley, M. F., Smith, J. C., and Cheng, X. (2009). Atomistic simulation of lignocellulosic biomass and associated cellulosomal protein complexes. In Nimlos, M. R. and Crowley, M. F., editors, *Computational Modeling in Lignocellulosic Biofuel Production*, page in print. ACS. 28

Pheatt, C. (2008). Intel threading building blocks. *J. Comput. Sci. Coll.*, 23(4):298–298. 22

Phillips, J. C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R. D., Kale, L., , and Schulten, K. (2005). Scalable molecular dynamics with NAMD. *J. Comput. Chem.*, 26:1781–1802. 7, 22, 24, 138

Pingali, S. V., Urban, V. S., Heller, W. T., McGaughey, J., O'Neill, H., Foston, M., Myles, D. A., Ragauskas, A., and Evans, B. R. (2010). Breakdown of cell wall nanostructure in dilute acid pretreated biomass. *Biomacromol.*, 11(9):2329–2335. 53, 80

Plimpton, S. (1995). Fast parallel algorithms for short-range molecular dynamics. *J. Comp. Phys.*, 117:1–19. 7, 24

Plimpton, S. (2004). Parallel fft package. http://www.sandia.gov/~sjplimp/docs/fft/README.html. 45

Polson, J. M. and Zuckermann, M. J. (2002). Simulation of short-chain polymer collapse with an explicit solvent. *J. Chem. Phys.*, 116(16):7244–7254. 52

Price, D. J. and Charles (2004). A modified tip3p water potential for simulation with ewald summation. *J. Chem. Phys.*, 121(20):10096–10103. 138

Pronk, S., Páll, S., Schulz, R., Larsson, P., Bjelkmar, P., Apostolov, R., Shirts, M. R., Smith, J. C., Kasson, P. M., van der Spoel, D., Hess, B., and Lindahl, E. (2013). GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinf.*, 29(7):845–854.

Appendix A is based on this.

7, 9

Pronk, S., Pouya, I., Rotskoff, G., Lundborg, M., Kasson, P., and Lindahl, E. (2014). Molecular simulation workflows as parallel algorithms: The execution engine of copernicus, a distributed high-performance computing platform. *J. Chem. Theory Comput.*, page Just Accepted Manuscript. 17

Pu, Y., Zhang, D., Singh, P., and Ragauskas, A. J. (2008). The new forestry biofuels sector. *Biofuels, Bioproduct and Biorefining*, 2:58–73. 54

Rocchi, C., Bizzarri, A. R., and Cannistraro, S. (1998). Water dynamical anomalies evidenced by molecular-dynamics simulations at the solvent-protein interface. *Phys. Rev. E*, 57(3):3315–3325. 72, 74

Ryckaert, J. P., Ciccotti, G., and Berendsen, H. J. (1977). Numerical-integration of cartesian equations of motion of a system with constraints - molecular-dynamics of n-alkanes. *J. Comput. Phys.*, 23(3):327–341. 16

Salmon, J. K., Moraes, M. A., Dror, R. O., and Shaw, D. E. (2011). Parallel random numbers: As easy as 1, 2, 3. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '11, pages 16:1–12, New York, NY, USA. ACM. 18

Samuel, R., Pu, Y. Q., Raman, B., and Ragauskas, A. J. (2010). Structural characterization and comparison of switchgrass ball-milled lignin before and after dilute acid pretreatment. *Appl. Biochem. Biotechnol.*, 162(1):62–74. 80

Sanbonmatsu, K. Y. and Tung, C. S. (2007). High performance computing in biology: Multimillion atom simulations of nanoscale systems. *J. Struct. Biol.*, 157(3):470–480. 24

Sarupria, S. and Garde, S. (2009). Quantifying water density fluctuations and compressibility of hydration shells of hydrophobic solutes and proteins. *Phys. Rev. Lett.*, 103(3):037803. 76, 82

Schulten, K., Phillips, J. C., Kalé, L. V., , and Bhatele., A. (2008). Biomolecular modeling in the era of petascale computing. In Bader, D., editor, *Petascale Computing: Algorithms and Applications*, pages 165–181. Chapman and Hall/CRC Press, Taylor and Francis Group. 44

Schulz, R., Lindner, B., Petridis, L., and Smith, J. C. (2009). Scaling of multimillion-atom biological molecular dynamics simulation on a petascale supercomputer. *J. Chem. Theory Comput.*, 5(10):2798–2808.

Chapter 3 is based on this.

125

Selig, M. J., Viamajala, S., Decker, S. R., Tucker, M. P., Himmel, M. E., and Vinzant, T. B. (2007). Deposition of lignin droplets produced during dilute acid pretreatment of maize stems retards enzymatic hydrolysis of cellulose. *Biotechnol. Progr.*, 23(6):1333–1339. 53, 80

Shirts, M. and Pande, V. S. (2000). Screen savers of the world unite! *Science*, 290(5498):1903–1904. 19

Smolin, N. and Winter, R. (2004). Molecular dynamics simulations of staphylococcal nuclease: Properties of water at the protein surface. *J. Phys. Chem. B*, 108(40):15928–15937. 70

Steinhauser, M. O. (2005). A molecular dynamics study on universal properties of polymer chains in different solvent qualities. part i. a review of linear chain properties. *J. Chem. Phys.*, 122(9):094901. 52

Stepanek, P., Konak, C., and Sedlacek, B. (1982). Coil-globule transition of a single polystyrene chain in dioctyl phthalate. *Macromol.*, 15(4):1214–1216. 52

Stillinger, F. H. (1973). Structure in aqueous solutions of nonpolar solutes from the standpoint of scaled-particle theory. *J. Solution Chem.*, 2:141–158. 52

Stockmayer, W. H. (1960). Problems of the statistical thermodynamics of dilute polymer solutions. *Makromolekulare Chemie*, 35:54–74. 52

Stone, J. E., Gullingsrud, J., and Schulten, K. (2001). A system for interactive molecular dynamics simulation. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, pages 191–194, New York, NY, USA. ACM. 18

Studer, M. H., DeMartini, J. D., Davis, M. F., Sykes, R. W., Davison, B., Keller, M., Tuskan, G. A., and Wyman, C. E. (2011). Lignin content in natural populus variants affects sugar release. *PNAS*, 108(15):6300–6305. 80

Sugita, Y. and Okamoto, Y. (1999). Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.*, 314:141–151. 17

Sun, S. T., Nishio, I., Swislow, G., and Tanaka, T. (1980). The coil-globule transition - radius of gyration of polystyrene in cyclohexane. *J. Chem. Phys.*, 73(12):5971–5975. 52

Svergun, D. I., Richard, S., Koch, M. H. J., Sayers, Z., Kuprin, S., and Zaccai, G. (1998). Protein hydration in solution: Experimental observation by x-ray and neutron scattering. *PNAS*, 95(5):2267–2272. 53

Takahashi, D. (2004). FFTE: A fast fourier transform package. http://www.ffte.jp. 45

Tarini, M., Cignoni, P., and Montani, C. (2006). Ambient occlusion and edge cueing for enhancing real time molecular visualization. 12(5):1237–1244. 31

ten Wolde, P. R. (2002). Hydrophobic interactions: an overview. *J. Phys. Condens. Matter*, 14(40):9445–9460. 52

ten Wolde, P. R. and Chandler, D. (2002). Drying-induced hydrophobic polymer collapse. *PNAS*, 99(10):6539–6543. 52

Tironi, I. G., Sperb, R., Smith, P. E., and van Gunsteren, W. F. (1995). A generalized reaction field method for molecular dynamics simulations. *J. Chem. Phys.*, 102(13):5451–5459. 26

Tribello, G. A., Bonomi, M., Branduardi, D., Camilloni, C., and Bussi, G. (2014). Plumed 2: New feathers for an old bird. *Comput. Phys. Commun.*, 185:604–613. 19

van der Spoel, D., Lindahl, E., Hess, B., Groenhof, G., Mark, A. E., and Berendsen, H. J. (2005a). GROMACS: Fast, flexible, and free. *J. Comput. Chem.*, 26(16):1701–1718. 9, 31, 55

van der Spoel, D., Lindahl, E., Hess, B., van Buuren, A. R., Apol, E., Meulenhoff, P. J., Tieleman, D. P., Sijbers, A. L., Feenstra, K. A., van Drunen, R., and Berendsen, H. J. (2005b). *GROMACS User Manual version 4.0*. 28, 29, 30

van der Spoel, D. and van Maaren, P. J. (2006). The origin of layer structure artifacts in simulations of liquid water. *J. Chem. Theory Comput.*, 2(1):1–11. 27, 35, 37, 47

van der Spoel, D., van Maaren, P. J., and Caleman, C. (2012). GROMACS molecule and liquid database. *Bioinf.*, 28(5):752–753. 19

Vermaas, J., Petridis, L., Qi, X., Schulz, R., Lindner, B., and Smith, J. C. (2015). Mechanism of lignin inhibition of enzymatic biomass deconstruction. *submitted*. 85

Walser, R., Hünenberger, P. H., and van Gunsteren, W. F. (2001). Comparison of different schemes to treat long-range electrostatic interactions in molecular dynamics simulations of a protein crystal. *Proteins: Structure, Function, and Genetics*, 43(4):509–519. 26, 47, 48

Wang, W. J., Kharchenko, S., Migler, K., and Zhu, S. P. (2004). Triple-detector gpc characterization and processing behavior of long-chain-branched polyethylene prepared by solution polymerization with constrained geometry catalyst. *Polymer*, 45(19):6495–6505. 66

Wennberg, C. L., Murtola, T., Hess, B., and Lindahl, E. (2013). Lennard-Jones Lattice Summation in Bilayer Simulations Has Critical Effects on Surface Tension and Lipid Properties. *J. Chem. Theory Comput.*, 9:3527–3537. 18

Wu, C. and Wang, X. H. (1998). Globule-to-coil transition of a single homopolymer chain in solution. *Phys. Rev. Lett.*, 80(18):4092–4094. 52

Wyman, C. E., Dale, B. E., Elander, R. T., Holtzapple, M., Ladisch, M. R., and Lee, Y. (2005). Coordinated development of leading biomass pretreatment technologies. *Bioresour. Technol.*, 96(18):1959 – 1966. 80

Yan, J. F., Pla, F., Kondo, R., Dolk, M., and McCarthy, J. L. (1984). Lignin .21. depolymerization by bond-cleavage reactions and degelation. *Macromolecules*, 17(10):2137–2142. 54

Yang, B. and Wyman, C. E. (2004). Effect of xylan and lignin removal by batch and flowthrough pretreatment on the enzymatic digestibility of corn stover cellulose. *Biotechnol. Bioeng.*, 86(1):88–98. 80

Yang, B. and Wyman, C. E. (2008). Pretreatment: the key to unlocking low-cost cellulosic ethanol. *Biofuels, Bioprod. and Biorefin.*, 2(1):26–40. 52

Yoluk, O., Brömstrup, T., Bertaccini, E. J., Trudell, J. R., and Lindahl, E. (2013). Stabilization of the GluCl ligand-gated ion channel in the presence and absence of ivermectin. *Biophys. J.*, 105:640–647. 19

Yu, Y. L., DesLauriers, P. J., and Rohlfing, D. C. (2005). SEC-MALS method for the determination of long-chain branching and long-chain branching distribution in polyethylene. *Polymer*, 46(14):5165–5182. 66

Zhou, Y. Q., Karplus, M., Wichert, J. M., and Hall, C. K. (1997). Equilibrium thermodynamics of homopolymers and clusters: Molecular dynamics and Monte Carlo simulations of systems with square-well interactions. *J. Chem. Phys.*, 107(24):10691–10708. 52

Zimm, B. H. and Stockmayer, W. H. (1949). The dimensions of chain molecules containing branches and rings. *J. Chem. Phys.*, 17(12):1301–1314. 66

Zoete, V., Cuendet, M. A., Grosdidier, A., and Michielin, O. (2011). SwissParam: A fast force field generation tool for small organic molecules. *J. Comp. Chem.*, 32(11):2359–2368. 19

Appendix

Appendix A

GROMACS 4.5: A high-throughput and highly parallel open source molecular simulation toolkit

This chapter is revised based on an paper with the same title as the chapter published in Bioinformatics, 2013, 29(7), 845-854 authored by Sander Pronk, Szilárd Páll, Roland Schulz, Per Larsson, Pär Bjelkmar, Rossen Apostolov, Michael R. Shirts, Jeremy C. Smith, Peter Kasson, David van der Spoel, Berk Hess, and Erik Lindahl.

Abstract

Motivation: Molecular simulation has historically been a low-throughput technique, but faster computers and increasing amounts of genomic and structural data have enabled large-scale automated simulation of many conformers, mutants, or ligands. At the same time, advances in scaling now make it possible to model complex protein motion and function in a manner directly testable by experiment. These

applications have a common need for fast and efficient software that can be deployed on massive scale in clusters, web servers, distributed computing, or cloud resources.

Results: Here, we present a range of new simulation algorithms and features developed over the last four years, leading up to the GROMACS 4.5 software package. The distribution automatically handles wide classes of biomolecules such as proteins, nucleic acids, and lipids and comes with all commonly used force fields for these molecules built-in. GROMACS supports several implicit solvent models optimized for screening and new free energy algorithms, and the software now employs multithreading for efficient parallelization even on low-end systems, including windows-based workstations. Together with hand-tuned assembly kernels and state-of-the-art parallelization, this provides very good performance and cost efficiency for high-throughput as well as massively parallel simulations.

Availability: GROMACS is open source and free software, and available from http://www.gromacs.org .

A.1 Introduction

Although molecular dynamics simulation of biomolecules is frequently classified as computational chemistry, the roots of the technique trace back to polymer chemistry and structural biology in the 1970s, where it was first used to minimize and relax structures to study local properties such as flexibility, distortion and stabilization on short time scales - essentially a tool to study the physics of local molecular properties. The field of molecular simulation has developed tremendously since then, and simulations are now routinely performed on multi-microsecond scale where it is possible to repeatedly fold small proteins, predict interactions between receptors and ligands, predict functional properties of receptors and even capture intermediate states of complex transitions, e.g. in membrane proteins. This classical type of single

long simulation continues to be very important. However, many studies increasingly rely on large sets of simulations, enabled in part by the ever-increasing number of structural models made possible by sequencing and structural genomics projects as well as new techniques to estimate complex molecular properties using thousands of shorter simulations. Mutation studies can now easily build models and run short simulations of hundreds of mutants, model-building web servers now offer automated energy minimization and refinement, and free energy calculations are increasingly being used to provide better interaction energy estimates than what is possible with docking. In these scenarios, molecular dynamics simulations can been seen at least as a medium-throughput method compared to traditional computational chemistry methods. Even more excitingly, with the exponential increase in available computational resources over recent years, there is potential for achieving truly high-throughput simulation performance in a not too distant future.

This development would not have been possible without significant research efforts in simulation algorithms, optimization, parallelization. The emergence of standardized packages for molecular modeling such as CHARMM, GROMOS, Amber, NAMD, and GROMACS has been important since they have helped commoditize simulation research, making the techniques available to life science application researchers who are not specialists in simulation development. All these packages have complementary strengths and profiles; for the GROMACS molecular simulation toolkit, one of our primary long-term development goals has been to achieve the highest possible simulation efficiency for the small- to medium size clusters present in our own research labs. Since computational resources are usually limited, it is often preferable to use throughput approaches with moderate parallelization rather than maximizing performance of an individual simulation. In recent years, we have also optimized the scaling of GROMACS to enable very long simulations when dedicated clusters or supercomputers are available for select critical problems.

Over the last four years since GROMACS 4, we have developed a number of new features and improvements that have led up to release 4.5 of the software

project and significantly improved both performance and efficiency for throughput and massively parallel applications. Many of the tasks that only a decade ago required exceptionally large dedicated supercomputing resources are now universally accessible, and sometimes they do not even require clusters. Below, we describe some of these features, including development to make the code fully portable and multithreaded on a wide range of platforms, features to facilitate high-throughput simulation, and not least more efficient tools to help automate complex simulations such as free energy calculations, with the long-term goal of commoditizing these techniques as well. High-end performance in GROMACS has also been improved with new decomposition techniques in both direct and reciprocal space that push parallelization further and which has made microsecond simulation timescales reachable in a week or two even for most large molecular systems without requiring excessive computational resources.

A.2 Results

A.2.1 An open source & free software framework for biomolecular simulation

The development of GROMACS was originally largely driven by our own needs for efficient modeling. However, in hindsight the decision to release the package as both open source and free software has been a tremendous advantage for the project. The codebase has gradually turned into a shared infrastructure with contributions from several labs world-wide, where every single patch and all code review is public as soon as they are committed to the repository. We explicitly encourage extensions and re-use of the code; as examples, GROMACS is used as a module to perform energy minimization in other structural bioinformatics packages (including commercial ones), it is available as a component from vendors that provide access to cloud computing resources, and we are happy to see some of the optimized mathematical functions

(such as inverse square roots) reused in other code. For the past few years, many Linux distributions have provided precompiled or contributed binaries of the package. These features per se do not necessarily say anything about scientific qualities, but we believe this open development model has been very efficient; compared to only ten years ago the project has evolved into a state where it is used everywhere from the smallest embedded processors to the largest supercomputers in the world, with applications ranging from genome-scale refinement of coarse-grained models to multi-microsecond simulations of membrane proteins or vesicle fusion.

A.2.2 Enabling efficient molecular simulation on desktop resources

Supercomputers are still very important for the largest molecular simulations, but large numbers of users rely on very modest systems for their computational needs. For many applications, one can even argue this is the most important target: many researchers rely on interactive tools, companies are hesitant to invest in expensive computational infrastructure, and there is an increasing focus on high-throughput studies, where a single calculation cannot use 50% of a cluster. Historically, this low-end regime has been the primary goal for GROMACS, and we have specifically focused efforts on achieving the highest possible efficiency on single nodes. GROMACS is designed for maximum portability, with external dependencies kept to a minimum and a fall-back internal library provided whenever possible. It has long been possible to build GROMACS on almost any Unix-based system (including many embedded architectures). In GROMACS 4.5, we have extended this further, making Microsoft Windows a fully-supported platform. This is obviously relevant for many researchers' desktops, but it is also critical for distributed computing projects where the software runs on participant-controlled computers, e.g. in the Folding@Home project. One of the main challenges in the last few years has been the emergence of multi-core machines. While GROMACS runs in parallel, it was designed to use MPI

communication libraries present on supercomputers rather than automatically using multiple cores. In release 4.5, we have solved this by designing a new internal "thread-MPI" interface layer that emulates the communication calls with multithreading and automatically uses every core available on a laptop or desktop for increased performance.

A.2.3 High-throughput simulation & modeling

As simulation software and computer performance has improved, molecular dynamics has increasingly been used for structure equilibration, sampling of models, or to test what effects mutations might have on structure and dynamics by introducing many different mutations and perform comparatively short simulations on multiple structures. While this type of short simulations might not look as technically impressive as long trajectories, we fundamentally believe it is a much more powerful approach. Since simulations build on statistical mechanics, a result merely seen in one long trajectory might as well be a statistical fluctuation that would never be accepted as significant in an experimental setting. In contrast, by choosing to perform e.g. 50 100-ns simulations instead of a single 5μs one it is suddenly possible to provide standard error estimates and quantitative instead of qualitative results from simulations. As discussed above, GROMACS has always been optimized to achieve the best possible efficiency using scarce resources (which we believe is the norm for most users), and version 4.5 has introduced several additional features to aid high-throughput simulation. All GROMACS runs are now automatically checkpointed and can be interrupted and continued as frequently as required, and optional flags have been added to enable binary reproducibility of trajectories. Since GROMACS is successfully used in a number of distributed computing projects where both CPU and storage hardware might be less controlled, `mdrun` now flushes all pending buffers after each file-writing step and also tries to flush file system cache when writing checkpoints. Since users are often working with data from thousands of simulations, we have

implemented MD5 signatures on checkpoint continuation files both to guarantee their integrity and to make sure the user does not append to the wrong file by mistake. These additional checks have further allowed us to enable continuation output appending by default; if a user simply specifies a standard continuation run in a job script this file can be executed over-and-over-again in a cluster queue until the entire simulation has completed, at which point it will appear as a single set of output just as if it came from a single execution. Hundreds or even thousands of smaller simulations can be started with a single GROMACS execution command to optimize use on supercomputers that favor large jobs, and each of these can be parallel themselves if advantageous. GROMACS also supports simulations running in several modern cloud computing environments; a concrete example of such usage is Amazon EC2 where virtual server instances can be started on demand. Since cloud computing usage is also billed by the hour, we believe the most instructive metric for performance and efficiency is to actually measure simulation performance in terms of the cost to complete a given simulation - for an example, see the performance section below.

A.2.4 Implicit solvent & knowledge-based modeling

In addition to the high-throughput execution model, there are a number of new code features developed to support modeling and rapid screening of structures. In previous versions GROMACS has not supported implicit solvent since it seemed of little point when it was slower than explicit water. This has changed with version 4.5, and the code now comes with very efficient implementations of the Still, HCT (Hawkins-Cramer-Truhlar) and OBC (Onufriev-Bashford-Case) (add citations) models for Generalized Born interactions based on tabulated interaction rescaling(Larsson and Lindahl, 2010). Together with manually tuned assembly kernels, implicit solvent simulations can reach performance in excess of a microsecond per day even on

standard CPUs. The neighbor-searching module has been updated to support grid-based algorithms even in vacuo - including support for atoms diffusing away towards infinity with maintained performance - and there are now also highly optimized kernels to compute all-vs-all interactions without cutoffs both for standard and Generalized Born interactions. The program now also supports arbitrary knowledge-based statistical interactions through atom-group specific tables both for bonded and nonbonded interactions. Constraints such as those used in refinement can be applied either to positions, atomic distances, or torsions, and there are several options for ensemble weighting of contributions from multiple constraints.

A.2.5 Strong scaling on massively parallel supercomputers

Despite the rapid emergence of high-throughput computing, the usage of massively parallel resources continues to be a cornerstone of high-end molecular simulation. Absolute performance is the goal for this usage too, but here it is typically limited by the scalability of the software. GROMACS 4.0 introduced a number of new features, including a completely new domain decomposition algorithm, but the performance was still limited by the less-than-ideal scaling of the particle-mesh Ewald (PME) implementation, in particular the single-dimensional decomposition of the fast Fourier transform (FFT) grids. For GROMACS 4.5, this has been solved with a new implementation of a two-dimensional, or "pencil" decomposition of reciprocal space. A subset of nodes are dedicated to the PME calculation, and at the beginning of each step the direct-space nodes send coordinate and charge data to them. Since direct space can be composed in all three dimensions, a number of direct-space nodes (typically 3-4) map onto a single reciprocal-space node. Limiting the computation of the 3D-FFT to a smaller number of nodes improves parallel scaling significantly, and the new pencil decomposition makes it much easier to automatically determine both real- and reciprocal-space decompositions of arbitrary systems to fit a given number of nodes. The automatic load balancing step of the domain decomposition has also

been improved, domain decomposition has been made to work even without periodic boundary conditions (important e.g. for implicit solvent) and GROMACS now comes with tools to aid in automatically tuning the balance between direct and reciprocal-space work. In particular when running in parallel over large numbers of nodes it is advantageous to move more work to real space (which essentially scales linearly) and decrease the reciprocal-space load to reduce the dimensions of the 3D-FFT grid (where the number of communication messages scales with the square of the number of nodes involved). The latest version of GROMACS also supports many types of multi-level parallelism; in addition to coding-level optimizations such as single-instruction multiple-data instructions and the multithreaded execution, GROMACS supports replica-exchange ensemble simulations where a single simulation can use hundreds of replicas that only communicate every couple of seconds, which makes it possible to scale even fairly small systems (e.g. a protein) to thousands of nodes. Finally, for the very largest systems comprising hundreds of millions of particles we now achieve true linear weak scaling for reaction-field and other non-lattice-summation methods. A lot of recent work has been invested in reducing memory needs and enabling parallel IO, and the code has been shown to successfully scale to over 150,000 cores.

A.2.6 Automated topology generation for wide classes of molecules & force fields

It was painfully obvious that the automated tools to generate input files were somewhat limited in earlier releases of GROMACS; few molecules apart from single-chain proteins worked perfectly. For version 4.5, the `pdb2gmx` tool has been reworked and we now support automatic topology generation for proteins, DNA, RNA, and many small molecules. Any number of chains and different molecule classes can be mixed, and they are automatically detected. The program provides several different options for how to handle termini and `HETATM` records in structures, and residue names and numbering from the input files are now

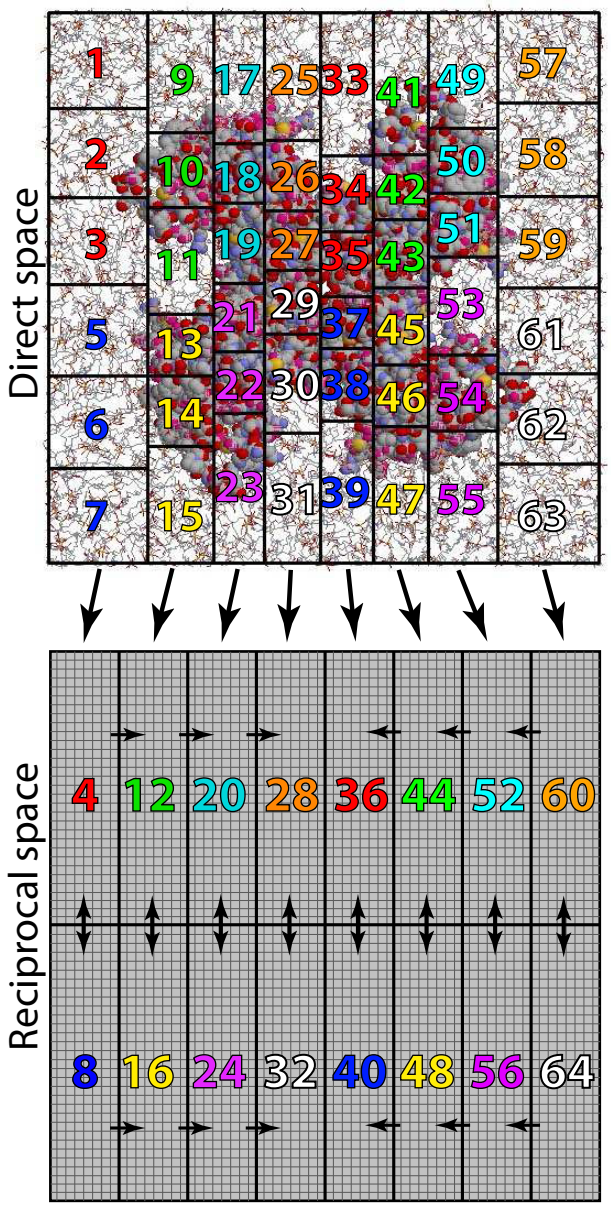


Figure A.1: 3D domain decomposition in real space combined with 2D pencil domain decomposition in reciprocal space.

maintained throughout the main simulation and analysis tools. With the most recent version, the package now comes with standard support for virtually all major point-charge force fields: GROMOS43a1, GROMOS43a2, GROMOS45a3, GROMOS53a5, GROMOS53a6, Encad, OPLS, OPLS-AA/L, CHARMM19, CHARMM27, Amber94, Amber96, Amber99, Amber99SB, AmberGS, Amber03, and Amber99SB-ILDN. To the best of our knowledge this range of forcefield support is unique and makes it straightforward to systematically compare the influence of the parameter approximations in biomolecular modeling. The code also provides naming translation data files to support all the conventions used in the different force fields.

A.2.7 A state-of-the art free energy calculation toolbox

Simulation-based free energy calculations provide a way to accurately include effects both of interactions and entropy, and accurately predict salvation and binding properties of molecules. It is one of the most direct ways that simulations can provide specific predictions of properties that can also be measured in wet-lab experiments. Both GROMACS and other packages have long supported slow-growth methods to calculate free energy differences when gradually changing the properties of molecules. The present release of the code provides an extensive new free energy framework based on Bennett Acceptance Ratio techniques focused on Hamiltonian differences for the system along an arbitrary coupling parameter λ. These differences are now calculated automatically on the fly in a simulation, rather than as a post-analysis step using gigantic trajectories, which makes it possible to perform the calculations e.g. in the cloud where the available storage and bandwidth is limited. Rather than manually defining how to modify/remove each molecule, the user can now simply specify that he/she wants to calculate the free energy of decoupling a particular molecule or group of atoms from the system and create input files. Given the set of output files from such a project, the code also comes with a new `g_bar` tool that automatically analyzes the statistical overlap, calculates the free energy of each component, and provides a

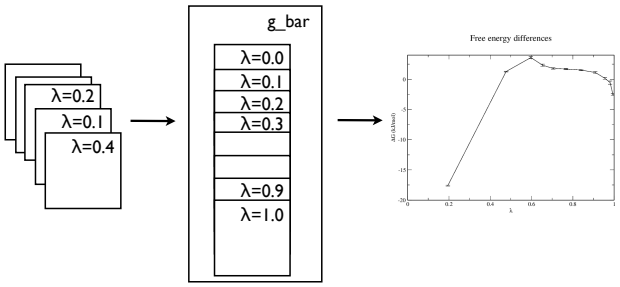


Figure A.2: 3D domain decomposition in real space combined with 2D pencil domain decomposition in reciprocal space.

finished curve of the free energy required to move from one end state to the other, including estimates of the standard error and sampling (Fig. X).

A.2.8 Other features

In addition to the larger development concepts covered here, several additional parts of GROMACS have been improved and extended for version 4.5.

- The core integration engine has been reworked and in addition to the simple leap-frog integrator we now support proper velocity-verlet integrators for fully reversible temperature and pressure coupling, with several new barostats and thermostats, including Nose-Hoover chains for ergodic temperature control and MTTK pressure control integrators. These are important for calculating accurate free energies, in particular for smaller systems or cases where factors such as pressure contributions will affect the result.

- The leap-frog integrator has also been rewritten on a symplectic Trotter form. This improves the accuracy of twin-range forces, but more importantly it enables correct multiple time-step integration of contributions from thermostats and barostats, which in turn makes it possible to not calculate global properties such as temperature every step to improve scaling.

- A new file-format plugin has been designed to allow GROMACS to read any trajectory or coordinate format supported by the VMD libraries without converting trajectories first, if the code is linked to the VMD libraries.

- The previously labor-intensive task of embedding and equilibrating membrane proteins in lipid bilayers has been automated with the new tool `g_membed` developed by Gerrit Groenhof. Given a membrane protein structure and an arbitrary bilayer (including ones with mixtures of lipid and/or other molecules), this tool virtually shrinks the membrane protein to a small axis and then expands it again while pushing lipids away over a few thousand steps. Lipids are removed based on overlap, and the tool has full support for asymmetrically-shaped proteins.

- Non-equilibrium simulation support has been extended to make it possible to pull any number of groups in arbitrary directions, and it is now also possible to apply torques in addition to forces.

- GROMACS now comes with extensive features for multi scale modeling built-in, including a QM/MM interface to a number of common quantum chemistry programs and algorithms, coarse-grained (CG) modeling with force fields such as MARTINI, and a highly efficient parallel implicit solvent algorithm that can all be used in combination.

- Normal-mode analysis can now be performed for extremely large systems through a new sparse-matrix diagonalization engine that also works in parallel, and even for PME simulations it is possible to perform the traditional non-sparse (computationally costly) diagonalization in parallel.

A.3 Performance

A.3.1 Scaling

For systems where absolute speed matters, the final simulation performance can be expressed as speed_per_core x number_cores x scaling_efficiency. We have thus aimed to improve both absolute performance per core and scaling efficiency in GROMACS. Recent enhancements in this respect include better Particle Mesh Ewald (PME) parallel decomposition. Choice of method for calculating long-range electrostatics can greatly affect simulation performance, and rather than simply optimizing the method that scales best, we have devoted effort to optimizing the method that is currently viewed as best practice in the field. By implementing a two-dimensional pencil node decomposition for PME and improving the dynamic load-balancing algorithms, we obtain linear scaling over large numbers of nodes for a set of benchmark systems that were selected as real world applications from our and others recent work. Scaling results are plotted in Fig. A.3 for a ligand-gated ion channel (Murail et al., 2011), a vesicle fusion simulation (Kasson et al., 2010), a virus capsid (Larsson et al., 2012), and a large methanol-water mixture (Schulz et al., 2009). To estimate real-world performance, we report scaling and performance results on two clusters: a Cray XE6 with a Gemini interconnect and a more commodity cluster with QDR Infiniband and less than full bisectional bandwidth. For all simulations except for the ion channel, we obtain strong linear scaling well over 1000 cores; for the ion channel, the linear scaling regime extends below 500 atoms/core. All of these benchmark simulations use PME long-range electrostatics; our tests with reaction field electrostatics show excellent linear scaling at even higher numbers of cores.

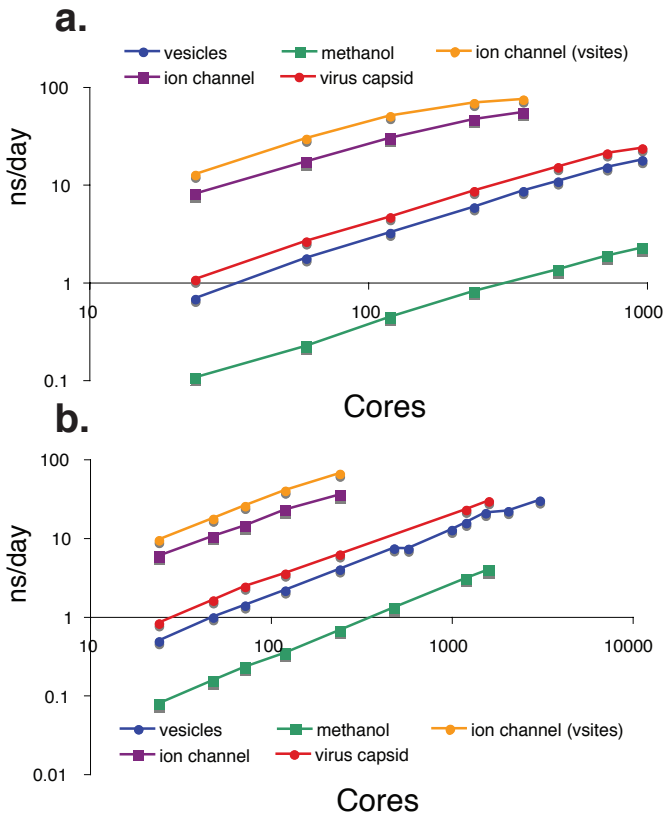


Figure A.3: Strong scaling of medium-to-large molecular systems. Simulation performance is plotted as a function of number of cores for a series of simulation systems. Performance data were obtained on two clusters: one that is thinly connected using QDR Infiniband but not full bisectional bandwidth and a more expensive Cray XE6 with a Gemini interconnect. In increasing order of molecular size: the ion channel with virtual sites had 129,692 atoms, the ion channel without virtual sites had 141,677 atoms, this virus capsid had 1,091,164 atoms, the vesicle fusion system had 2,511,403 atoms, and the methanol system had 7,680,000 atoms.

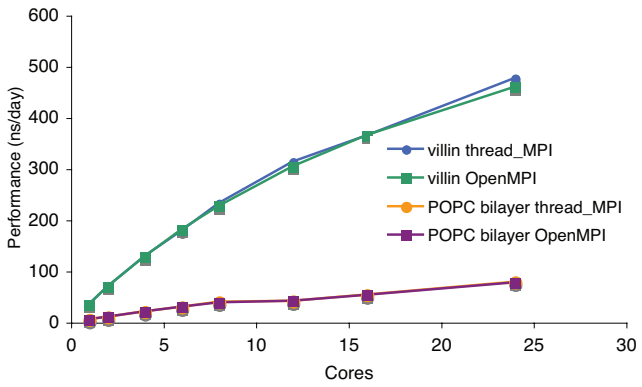


Figure A.4: An efficient parallel implementation on pthreads and Windows threads. Simulation performance is plotted as a function of number of cores for Gromacs using the thread_MPI library and compared to Gromacs using OpenMPI on the same system. Both sets of simulations were run using a single node with 24 AMD 8425HE compute cores running at 2.66 GHz. Performance is nearly identical between the two parallel implementations. Data are plotted for the villin and POPC bilayer benchmark systems.

A.3.2 Single-node parallelization

GROMACS 4.5 implements parallelization at a low level through SIMD and at a high level through MPI, although ongoing efforts add an intermediate level of OpenMP parallelization. This provides good scaling at high core counts but adds complexity to code deployment for small installations. We have therefore added a threads-only implementation of MPI primitives that allows single-node parallelization of GROMACS using either pthreads or Windows threads without additional dependencies. Scaling of the thread_MPI implementation is plotted in Fig. A.4; the scaling behavior is near-identical to OpenMPI on a single node, which is a good comparison because single-node OpenMPI will use shared memory parallelization when advantageous. The advantage of the GROMACS thread_MPI implementation is that is is lightweight, reduces build complexity, and works on a wide variety of systems including Linux, OS/X, and Windows. This had also greatly facilitated large-scale deployment of parallel GROMACS simulations on architectures such as Folding@Home.

A.3.3 Throughput simulations

As modern computers have increased in processing power, simulations that used to require supercomputers become tractable on small clusters and even single machines. This has two important consequences: moderate-size simulations become accessible to non-specialists without major allocations of supercomputing resources, and it becomes possible to run many simulations at once to perform moderate-throughput computation on different reaction conditions, mutants of a protein, or small-molecule ligands. To illustrate both of these, we have benchmarked GROMACS on Amazon EC2 instances. The cloud-computing market gives access to relatively capable machines and good burst capacity to thousands of cores or more. With the thread_MPI parallelization in GROMACS, simulation performance is quite good on single nodes. To emphasize the general accessibility of performing these simulations, we have also plotted the cost per microsecond of simulation in each of these systems in Figure A.5. In 1998, Duan and Kollman published a one-microsecond simulation of the villin headpiece, a landmark computational achievement at the time. That simulation required 4 months of supercomputer time; we can perform the same simulation in under a week on a single EC2 node at a cost of $11.28, bringing this within range of a student project. Equivalently, screening of hundreds of mutants becomes feasible even without large dedicated resources.

A.4 Conclusions & Outlook

Improvements in processor power, simulation algorithms, and new computing paradigms are opening a new frontier for molecular dynamics simulation where many simulations of a moderate-sized system are now tractable. This enables a fundamental change in the way we approach molecular simulation as a tool. The traditional use of molecular dynamics can be thought of as probing the physical consequences of a given starting protein sequence, ligand, and structure. Now, given

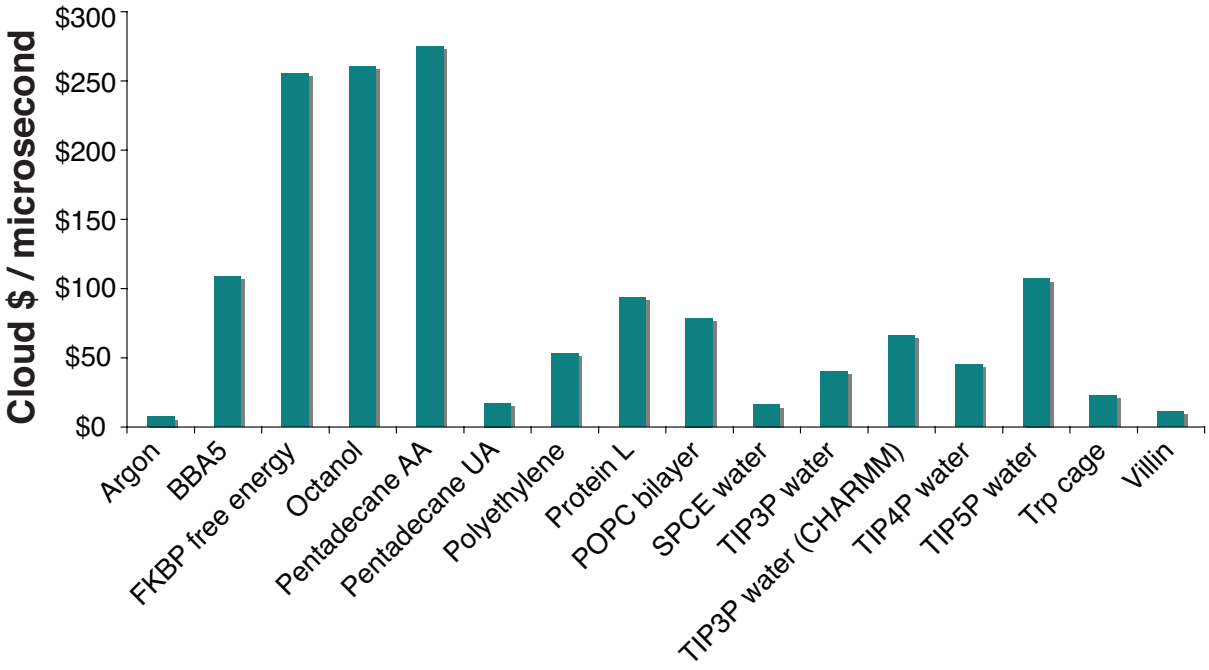


Figure A.5: Cost efficiency of Gromacs on small molecular systems. Cost per microsecond is plotted for a series of small molecular systems when run on a single 8-core node of Amazon EC2. The ready availability of cloud compute instances enables extremely cost-efficient simulation using individual nodes. Flexible capacity and low cost make a wide range of high-throughput applications possible. Gromacs has been optimized both for single-node compute speed and for scaling on large clusters.

an ensemble of 50 candidate models (as might be generated from NMR of a flexible or underdetermined complex), we can evaluate the relative probability of the models and use simulation to define the space explored by the models. We can also do mutant scans–the computational equivalent of combinatorial mutagenesis. Soon, approaches such as random walks in sequence space or ligand scaffold space will become routinely tractable. These new capabilities will demand a different approach to simulation–in addition to the underlying physics- and chemistry-based methodology, scientists will need to devote more attention to statistical sampling and can leverage the benefits of classically informatics techniques such as randomized search algorithms and network flow theory. We thus believe that advances in molecular dynamics simulation are transforming computational biophysics into more of a data-based science.

GROMACS 4.5 has continued to improve several key aspects of the simulation package: performance, support of a wide range of force field parameters and simulation methods, portability, and analysis tools. Of particular note, we have added better decomposition methods for scaling to many nodes, better single-node parallelization, support for additional force fields and implicit solvent methods, and free-energy analysis tools. Over the next couple of years, we expect the high-throughput trend to become increasingly accentuated: Despite massively increased computational power, researchers have been reluctant to merely push longer simulations. Instead of extending membrane proteins simulations to a single 5-10 microsecond trajectory, most current publications rather use the same amount of total computing time for a whole set of shorter simulations to provide statistics. Fundamentally, we believe this is a scientifically sound development, and one that is likely to move biomolecular simulation and modeling from compute-centric to data-centric approaches more similar to other methods used in bioinformatics.

Appendix B

Supporting information for chapter 3

B.1 FFT

The new 3D FFT implementation is a parallelized FFT that uses MPI for communication and the FFTW library (Frigo and Johnson, 2005) for the 1D-FFT using stencil decomposition as described in Ref. Eleftheriou et al. (2005). The performance of the FFT was optimized and improved compared to other current parallel FFT libraries, by combining a number of steps: (i) Use of the SIMD-optimized FFT available with FFTW 3.x was used for the 1D FFT. (ii) Use of the fftw_mpi_plan_many_transpose function for the communication. (iii) Support of data sizes with box lengths not multiples of the number of cores (without the speed penalty of MPI_Alltoallv). (iv) The use of real numbers and single precision to reduce the amount of data. The details will be described in an upcoming paper.

GROMACS has task parallelism implemented for PME (Hess et al., 2008), which improves the scaling. To compare the performance of RF with that achievable with PME, we benchmarked the FFT on the number of CPUs that would be dedicated for the PME part of an MD simulation. The benchmarks were run on 3 out of the

8 cores of each XT5 node, so as to simulate the situation in which the other cores perform the non-PME force calculations.

The Inset to 3.11 shows the time for four different implementations on the Cray XT4 for real→ complex 128x128x128 FFT on 1024 cores. For FFTW the maximum possible 128 cores are used. P3DFFT is not shown because a newer version has been made available since the comparison.

The details of how the weak scaling was obtained of complex-to-complex FFT on Cray XT5 (with FFT implemented as described above), shown in 3.11, are as follows. The data sizes are 32x32x32, 64x64x64, 128x128x128, 588x128x128, 480x480x480 for 16, 128, 1024, 4704, 38400 cores, respectively. The 3.3 million atom system requires the 588x128x128 FFT. 480x480x480 is the largest cube that currently fits onto the XT5 with 2048 data per core and 3 cores used per node. The 480x480x480 size FFT is faster on 38400 cores than for 54000 cores since the size is better divisible for 38400 cores. Therefore 38400 cores were used. Times for the local transpose, the 1D FFT and the two global transposes are stacked in 3.11. The local time is the sum of the two transposes required for one 3D FFT step. Similarly, a 1D-FFT step requires three FFT computations and the 1D FFT time quoted is the sum of the three times.

B.2 Generation of Topologies

The creation of branched polymer initial starting configurations and topology files is not possible using the tools currently available to GROMACS. Since lignin is a branched polymer, the topology was generated in CHARMM (Brooks et al., 1983) and the resulting "Protein-Structure-File" converted into the GROMACS topology format using a locally-modified version of psfgen (Gullingsrud et al., 2006). This conversion required several steps. First, electrostatics methods require the careful definition of "charge-groups" in the topology files so as to avoid cut-off based artefacts. For this, atoms are grouped together such that the total charge of the group is neutral and monopole cut-off effects are thus avoided. Second, GROMACS requires both the

atomic charge-group and monomer number to be consecutive in the topology. This poses a problem in branched polymers such as lignin, where charge groups can involve more than one monomer. Therefore, it was necessary to use the same residue number for a whole polymer and to reorder atoms so as to make charge-groups consecutive in the topology file. Finally, the charge-groups were made neutral: originally, this was not the case because CHARMM split charge-groups spanning several monomers. The force field parameter files were converted into GROMACS format using a script.

B.3 Comparison of Simulations with Different Electrostatic Methods

This section contains further material concerning the validation of the use of the RF method to treat electrostatic interactions. Exactly as in Section 3.3.1, various physical properties are calculated using the three electrostatics methods: PME, RF and Shift. In all cases presented in this section, there is excellent agreement between the results obtained from all three methods. The physical measures compared are: the root-mean-squared-deviation in B.1 and the PMFs of the Φ- and Ψ- dihedrals (for their definition see 3.7) in B.3 and B.4.

Root Mean Square Displacement In B.1 the root-mean squared deviation between the initial (crystal) structure of the cellulose fibril and its structure at each frame of the simulation is plotted for the three electrostatics methods. The RF and Shift methods reproduce the PME-derived results well.

Principal Component Analysis Principal component analysis (PCA) is a method that extracts the dominant modes in the motion of the molecule from an MD trajectory (Amadei et al., 1993). These modes are obtained by diagonalizing the atomic displacement covariance matrix and then sorting the eigenvalues in order of decreasing value. The first few eigenvectors display the highest fluctuations and

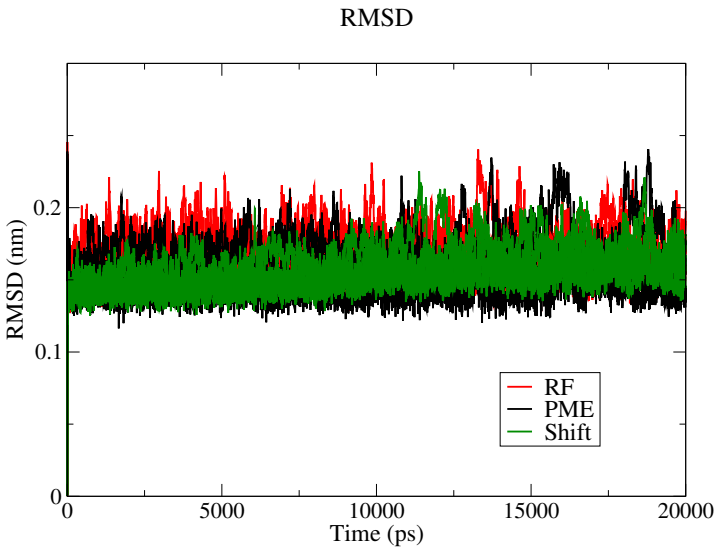


Figure B.1: Root-mean-squared deviation between the crystal structure of the cellulose fibril and the structure at each frame of the MD trajectory.

correspond to collective motions of groups of atoms that determine the essential dynamics of the biomolecule. B.2 shows the eigenvalues of the covariance matrix for the C1 chain of the fibril, with the most dominant modes displaying larger eigenvalues. Here, both the RF and Shift methods reproduce well the PME-derived amplitudes of the most dominant modes.

Cellulose Dihedral Angles Potential of mean force plots for the three cellulose dihedral angles under study were generated from the distribution statistics. Section 3.3.1 of the main text already presented the PMF results for the ω-angle. The Ψ and Φ PMF graphs are shown in B.3 and B.4, respectively. The results for the average angle and PMF minima agree well with the experimentally-observed values (French and Johnson, 2004). The agreement between PME and RF is good, while Shift shows slight deviations in PMF minima positions. However, in contrast to the ω angle, the Ψ and Φ angles are relatively insensitive to changes in electrostatic treatment.

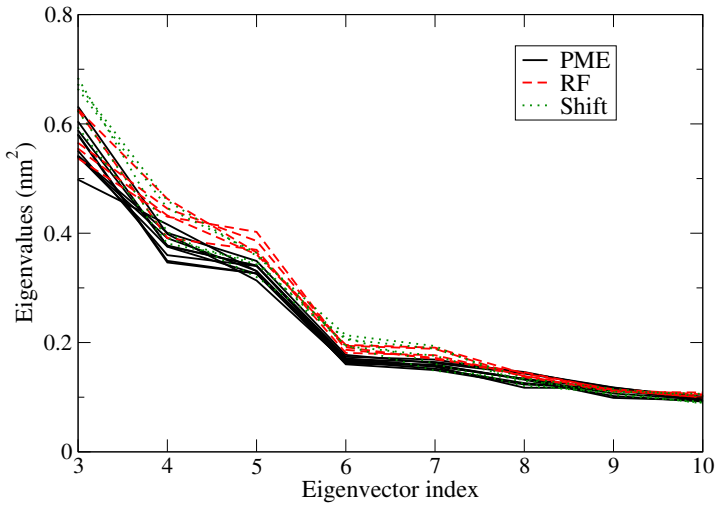
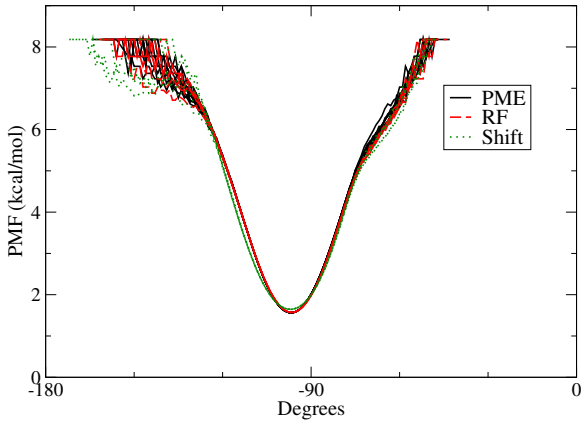
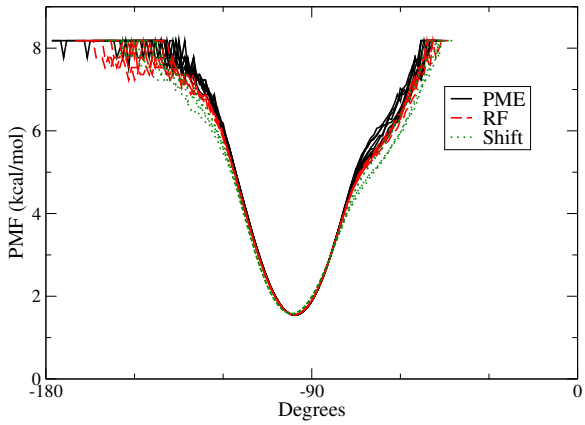


Figure B.2: Principal component analysis of one cellulose chain. The largest collective vibrations in the cellulose are indexed in the x-axis and the associated eigenvalues are shown in the y-axis.

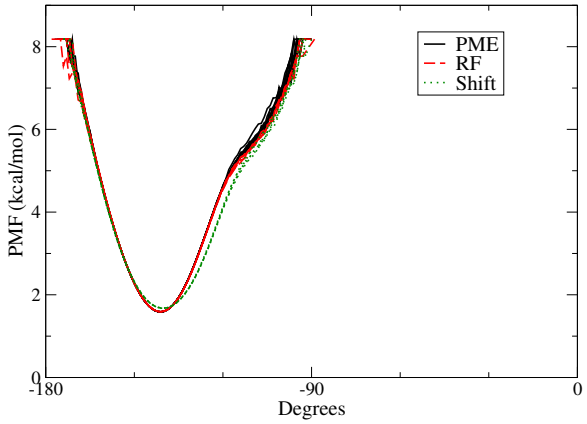


(a) Origin Chain

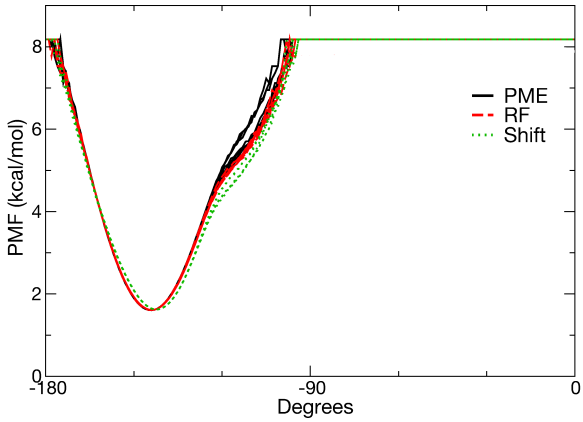


(b) Center Chain

Figure B.3: Potentials of mean force for the Φ dihedral (O5-C1-O1-C4*) (a) results from all 36 origin chains and (b) results from all 36 center chains.



(a) Origin Chain



(b) Center Chain

Figure B.4: Potentials of mean force for the Ψ dihedral (C1-O1-C4*-O5*) (a) results from all 36 origin chains and (b) results from all 36 center chains.

Table B.1: Energy comparison for lignin and cellulose. Total energy in kJ/mol for seven lignin dimers (infinite cut-off) and a 160-monomer cellulose chain (PME) with GROMACS (G), CHARMM (C) and NAMD (N).

energy type	lignin(C)	lignin(G)	cellulose(N)	cellulose(G)
bond	165.90	165.90	25552.7	25552.7
angle	257.35	257.35	4127.4	4127.5
dihedral	1658.93	1658.93	-17293.3	-17293.3
Coulomb	2276.17	2276.12	45622.0	45621.3
VDW	728.74	728.74	5413.4	5415.0

B.4 Cross-application Comparison

Since the present paper reports on the first use of the cellulose and lignin force fields in GROMACS, a cross-application comparison was performed, in order to examine whether the same results are obtained when the same simulation is run with NAMD and with GROMACS.

Methods. Cross-application reproducibility was examined via two simulations performed with NAMD (Phillips et al., 2005) and GROMACS 4.0.4, using identical protocols as in Section 3.2.1. For the comparison of GROMACS with NAMD the settings in Section 3.2.1 were modified so as to allow the same settings for both programs. For this a cubic box was constructed with 54272 mTIP3P (Price and Charles, 2004) water molecules, totaling 223404 atoms. All bonds involving hydrogen were constrained (Order: 4, Iterations: 1). The pressure coupling during equilibration was anisotropic and Berendsen pressure (τ = 4ps) and temperature coupling (τ = 0.1ps) was used for the equilibration and production.

B.4.1 Consistency in energies between CHARMM, NAMD and GROMACS

An MD code calculates atomic interactions based on the description given by the force field and the topology. The calculated interaction energies are therefore an indicator

for the correct implementation and interpretation of the force field, as well as being sensitive to errors in the topology file. B.1 shows energy values computed for a single chain of cellulose and the sum of the 7 lignin dimers with the different possible linkages (B.2 for individual energies). Also, the energy of a lignin polymer was compared (not shown). The computed interaction energies agree within numerical accuracy for the different MD codes. The van der Waals (VDW) energy for cellulose differs slightly more than the numerical accuracy. However, this is because charge groups were used with GROMACS but not with NAMD. The charge-groups affect the neighbor list and thus also have a slight effect on the VDW energy.

B.4.2 Dynamic Properties

In the same way as described in Section 3.3.1 of the main text, the GROMACS and NAMD trajectories were analyzed for all prior properties discussed above *i.e.*: Kirkwood function, total dipole, RMSF, RMSD, PCA and PMF of the three dihedrals. The results are shown in B.5 to B.12. For all properties the NAMD and GROMACS results agree very well.

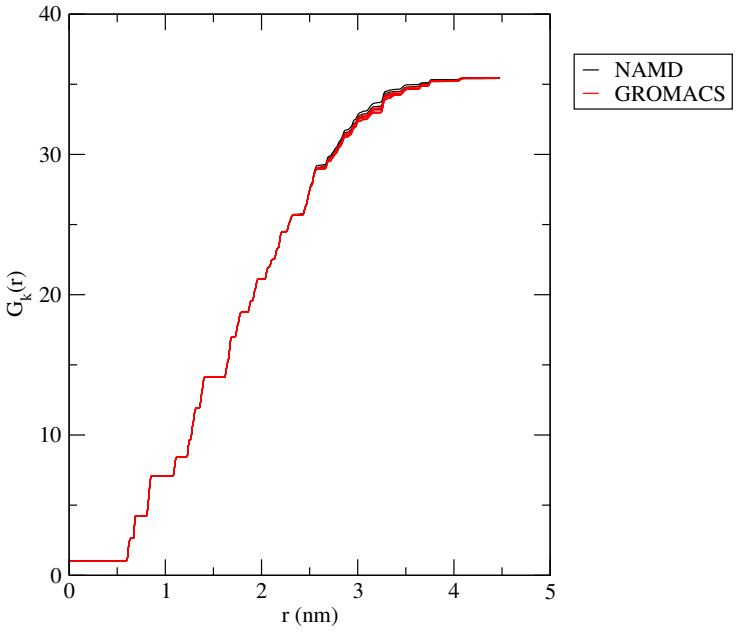


Figure B.5: Cross-application comparison of the distance-dependent Kirkwood factor (eq. 3.2).

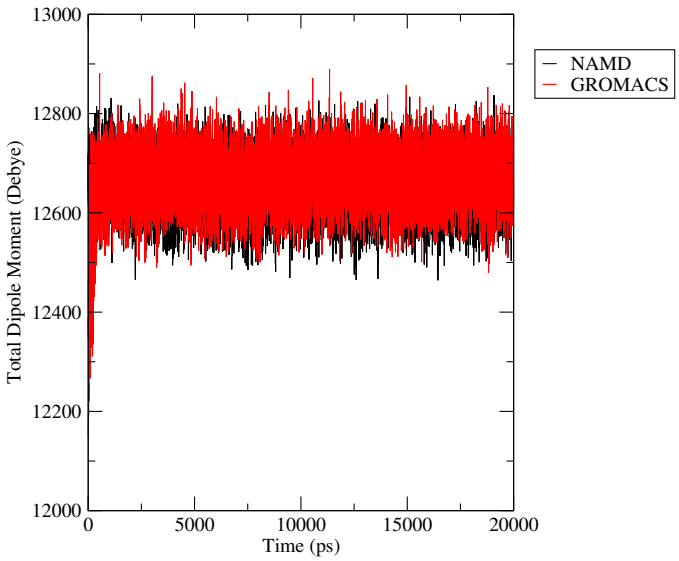


Figure B.6: Cross-application comparison of the total dipole moment of the cellulose fibril.

Table B.2: Energy comparison for 7 lignin dimers, with monomers connected with 55, b5r, ao4r, bo4r, bo4l, ao4l, bo4r, b5l linkages. Total energy in kJ with infinite cut-off.

55	CHARMM	GROMACS	b5r	CHARMM	GROMACS
bond	30.322	30.322	bond	21.590	21.590
angle	32.217	32.217	angle	56.104	56.104
dihedral	274.038	274.038	dihedral	278.490	278.490
Coulomb	414.668	414.658	Coulomb	243.658	243.652
vdw	141.437	141.437	vdw	103.068	103.068

ao4r	CHARMM	GROMACS	bo4l	CHARMM	GROMACS
bond	21.791	21.791	bond	23.433	23.433
angle	24.648	24.648	angle	27.204	27.204
dihedral	205.362	205.362	dihedral	206.794	206.794
Coulomb	362.428	362.419	Coulomb	339.255	339.248
vdw	96.511	96.511	vdw	94.854	94.854

ao4l	CHARMM	GROMACS	bo4r	CHARMM	GROMACS
bond	22.330	22.330	bond	24.471	24.471
angle	24.850	24.851	angle	27.854	27.854
dihedral	206.152	206.152	dihedral	217.263	217.263
Coulomb	351.013	351.004	Coulomb	321.090	321.082
vdw	97.065	97.065	vdw	90.264	90.264

b5l	CHARMM	GROMACS
bond	21.961	21.961
angle	64.470	64.471
dihedral	270.830	270.830
Coulomb	244.059	244.053
vdw	105.543	105.543

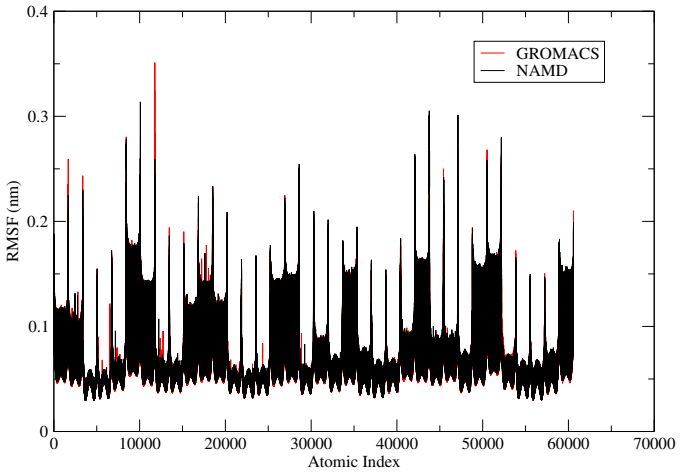


Figure B.7: Cross-application comparison of the RMSF for all atoms in the cellulose fibril (atomic index on x-axis).

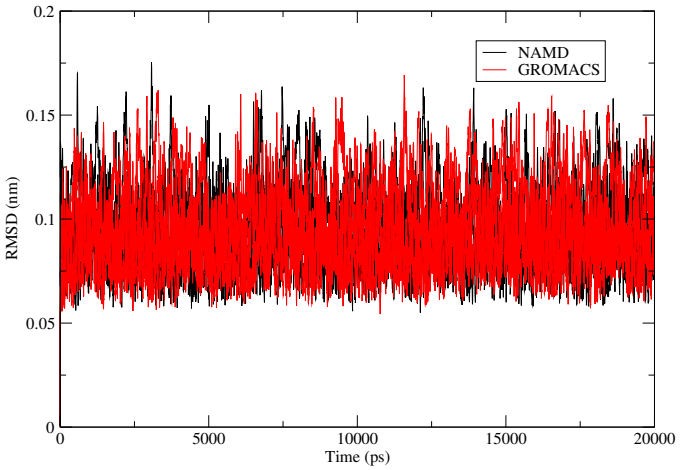


Figure B.8: Cross-application comparison of the root-mean-squared deviation between the crystal structure of the cellulose fibril and the structure at each frame of the MD trajectory.

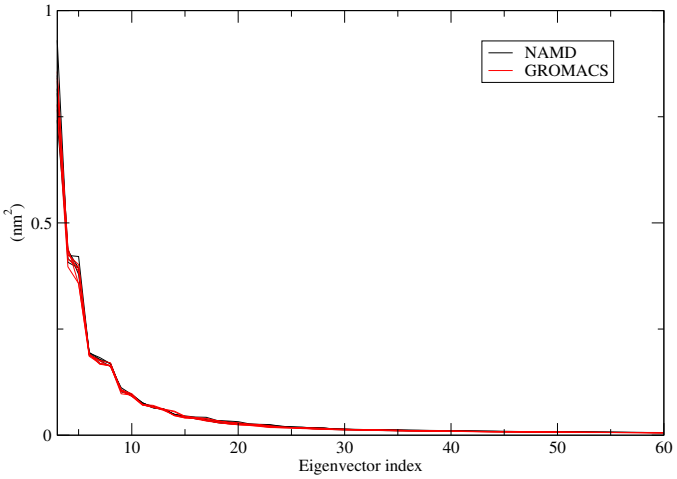
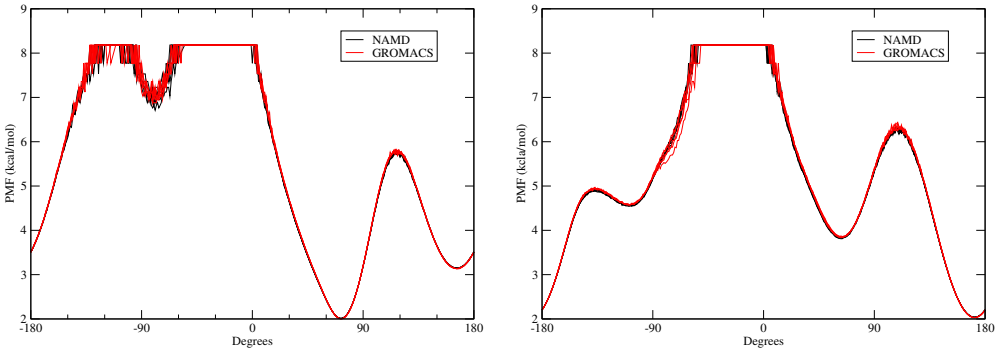
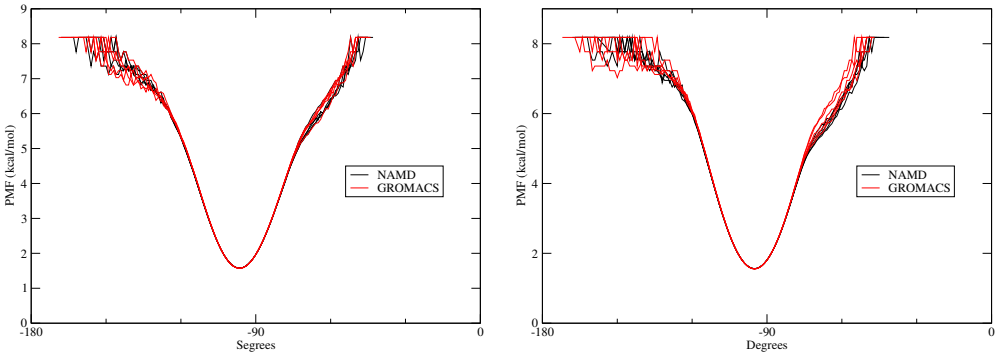


Figure B.9: Cross-application comparison of the Principal Component Analysis of one cellulose chain. The eigenvectors are indexed on the x-axis and the associated mean squared fluctuation of each mode is shown on the y-axis.



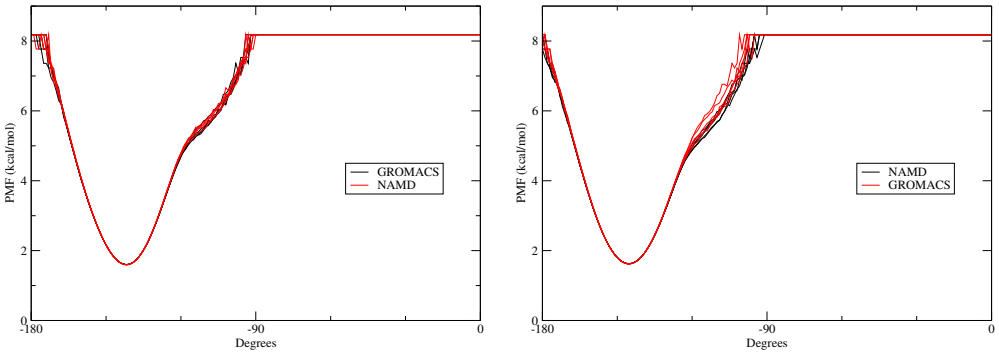
(a) Origin Chain **(b)** Center Chain

Figure B.10: PMF for the primary alcohol dihedral ω =O6-C6-C5-C4: (a) results from all 36 origin chains and (b) results from all 36 center chains.



(a) Origin Chain **(b)** Center Chain

Figure B.11: PMF for the Φ dihedral Φ =O5-C1-O1-C4*: (a) results from all 36 origin chains and (b) results from all 36 center chains.



(a) Origin Chain **(b)** Center Chain

Figure B.12: PMF for the Ψ dihedral Ψ =C1-O1-C4*-O5*: (a) results from all 36 origin chains and (b) results from all 36 center chains.

Appendix C

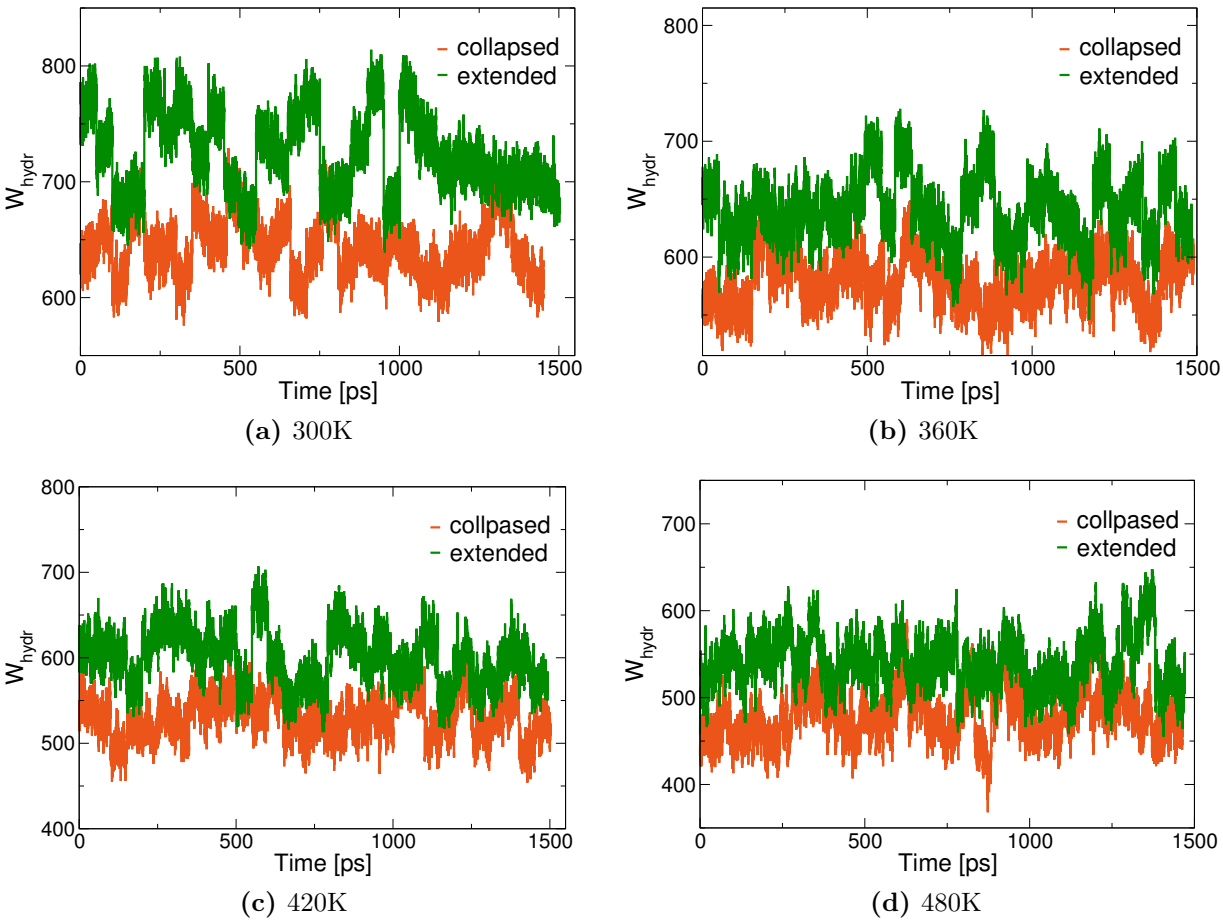
Supporting information for chapter 4

C.1 Methods Summary

The analysis of the collapse transition is based on comparison of the thermodynamic properties of collapsed and extended states as follows. The enthalpy contribution is derived from the interaction energies corresponding to R_{ext} and R_{col} in Figure 4.6. The translational and rotational entropy contributions of hydration water were derived from ten 20ps simulations, five starting from extended and the other five from collapsed lignin configurations. The entropy contribution from water density fluctuations was derived from sixty 50ps simulations, 30 from collapsed and 30 from extended states.

C.2 Structural Properties of the Collapsed and Extended Lignins

Table C.1 summarizes the structural properties of the 6 simulations that were used to investigate the thermodynamic origin of lignin collapse at 300K. Figure C.1 shows



(a) 300K **(b)** 360K

(c) 420K **(d)** 480K

Figure C.1: Number of water molecules in the hydration shell as a function of time at (a) 300K, (b) 360K, (c) 420K and (d) 480K. Data from 30 50ps trajectories are appended to yield the 1.5ns plot.

Table C.1: Comparison of radius of gyration (R), solvent-accessible surface area (A) and number of hydration water molecules of collapsed and extended lignin structures at various temperatures. Values are averages over 30 50ps trajectories.

	R_{col} [nm]	A_{col} [nm^2]	W_{col}	R_{ext} [nm]	A_{ext} [nm^2]	W_{ext}
300K	1.41±0.01	72.8±1.8	644±24	1.63±0.02	80.7±3.9	725±34
360K	1.40±0.02	72.4±2.4	580±22	1.60±0.03	79.2±3.4	640±27
420K	1.42±0.02	76.0±3.3	532±26	1.61±0.05	84.6±4.0	600±32
480K	1.44±0.04	82.5±4.3	479±28	1.64±0.04	91.6±4.2	540±30

the number of water moelcules in the hydration shell of lignin used to determine water compressibility.

C.3 Temperature Dependence of Lignin Structure

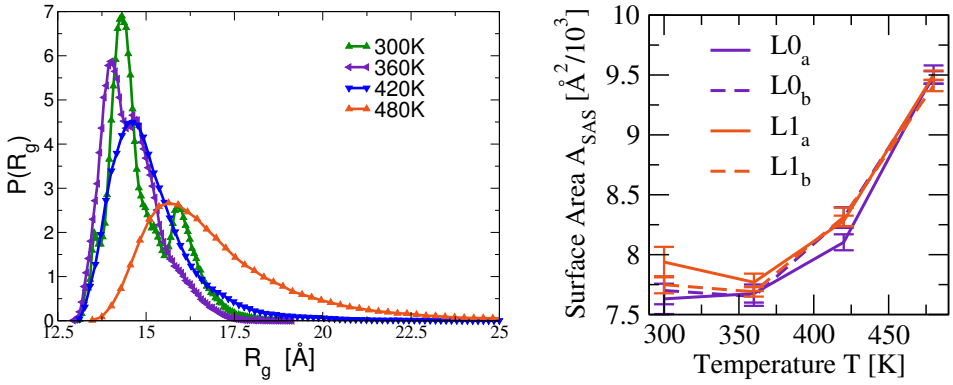


Figure C.2: Temperature dependence of the lignin structural properties. (a) The probability distribution of the radius of gyration, R_g. (b) Solvent accessible surface area, see the legend of Figure 4.1 for error estimates.

C.4 Analytic Theory for Effect of Branching on Polymer Size

In what follows we modify the Zimm-Stockmayer (ZS) theory[6]8 to render it applicable to polymers in poor solvents, such as lignin in water. The key assumption here is that the distance D_{ij} between monomers i and j is given by

$$D_{ij} = bN^{1/3}, \tag{C.1}$$

where i and j are separated by a linear segment of $(N-1)$ monomers and b is the Kuhn length of a monomer, a measure of the monomer average size. By further assuming isotropy in space and long chains ($N_{tot} >> 1$), the R_g of a branched chain can be in principle calculated for any degree of branching:

$$R_g^2 = \frac{1}{2N_{tot}^2}\left\langle \sum_{i,j=1}^{N_{tot}} D_{ij}^2 \right\rangle. \tag{C.2}$$

Our discussion is confined to a polymer with one branch point, with its three arms each consisting of N_1, N_2 and N_3 monomers, respectively ($N_{tot} = N_1 + N_2 + N_3$). Let the branch point be positioned at the monomer with index $k = 1$, the first arm comprising monomers $k = [2, N_1]$, the second arm $k = [N_1+1, N_1+N_2]$ and the third $k = [Ni + N_2 + 1, N_{tot}]$. In the limit $N_1, N_2, N_3 >> 1$, the r_g of a one-branch point polymer in Eq. C.2 is readily decomposed into the following sum:

$$R_{g,1}^2 = A_1 + A_2 + A_3 + 2(A_4 + A_5 + A_6), \tag{C.3}$$

where,

$$
\begin{aligned}
A_1 &= \frac{1}{2N_{tot}^2}\int_0^{N_1} di \int_0^{N_1} dj|i-j|^{2/3} = \frac{9}{80N_{tot}^2}N_1^{8/3} && \text{(C.4)}\\
A_2 &= \frac{1}{2N_{tot}^2}\int_{N_1}^{N_1+N_2} di \int_{N_1}^{N_1+N_2} dj|i-j|^{2/3} = \frac{9}{80N_{tot}^2}N_2^{8/3}\\
A_3 &= \frac{1}{2N_{tot}^2}\int_{N_1+N_2}^{N_1+N_2+N_3} di \int_{N_1+N_2}^{N_1+N_2+N_3} dj|i-j|^{2/3} = \frac{9}{80N_{tot}^2}N_3^{8/3}\\
A_4 &= \frac{1}{2N_{tot}^2}\int_0^{N_1} di \int_{N_1}^{N_1+N_2} dj|i+j-N_1|^{2/3} =\\
&= \frac{9}{80N_{tot}^2}\left[(N_1+N_2)^{8/3} - N_1^{2/3} - N_2^{2/3}\right] && \text{(C.5)}\\
A_5 &= \frac{1}{2N_{tot}^2}\int_0^{N_1} di \int_{N_1+N_2}^{N_1+N_2+N_3} dj|i+j-N_1-N_2|^{2/3} =\\
&= \frac{9}{80N_{tot}^2}\left[(N_1+N_3)^{8/3} - N_1^{2/3} - N_3^{2/3}\right] && \text{(C.6)}\\
A_6 &= \frac{1}{2N_{tot}^2}\int_{N_1}^{N_1+N_2} di \int_{N_1+N_2}^{N_1+N_2+N_3} dj|i+j-2N_1-N_3|^{2/3} =\\
&= \frac{9}{80N_{tot}^2}\left[(N_3+N_2)^{8/3} - N_3^{2/3} - N_2^{2/3}\right]. && \text{(C.7)}
\end{aligned}
$$

The squared radius of gyration of an unbranched collapsed polymer is $R_{g,0}^2 = 9b^2N_{tot}^{2/3}/80$. Hence, by substituting Eqs. C.4 and C.7 into Eq. C.3 for the $R_{g,1}^2$ of a collapsed polymer with one branch point, we obtain the theoretical prediction of their ratio

$$
\begin{aligned}
g = \frac{R_{g,1}^2}{R_{g,0}^2} &= \left(\frac{N_1+N_2}{N_{tot}}\right)^{8/3} + \left(\frac{N_2+N_3}{N_{tot}}\right)^{8/3} + \left(\frac{N_3+N_1}{N_{tot}}\right)^{8/3}\\
&\quad - \left(\frac{N_1}{N_{tot}}\right)^{8/3} - \left(\frac{N_2}{N_{tot}}\right)^{8/3} - \left(\frac{N_3}{N_{tot}}\right)^{8/3}. && \text{(C.8)}
\end{aligned}
$$

Similarly to the ZS theory for polymers in ideal solvents, Eq. C.8 also predicts $g < 1$. However Eq. C.8 also indicates that branching reduces the size of isotropic collapsed polymers less than isotropic Gaussian chains. For a star polymer, where $N_1 = N_2 = N_3 = (N_{tot}/3)$ Eq. C.8 gives $g = 0.86$, whereas with the ZS theory $g_{ZS} = 0.77$.

C.5 Scaling Properties of Branched Lignins

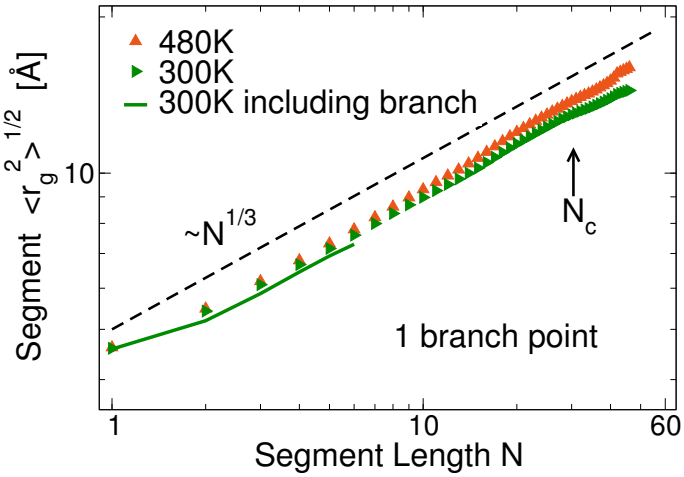


Figure C.3: Root mean square of the radius of gyration $r_g(N)$ of a polymer segment comprising $(N+1)$ monomers of the ensemble of polymers with one branch point. Also shown, as a green solid line, are data from short ($(N \leq 6)$) segments that include the branch point. The dashed black line is a $\sim N^{0.33}$ power-law function and all plots are time averages over the last 50ns of the ensemble of 20 MD trajectories. The error bars are the standard deviations of the ensemble distribution.

C.6 Correlation of R_g and Δ

Figures C.4(a),(b) demonstrates the strong correlation between the R_g and the asphericity, δ of a lignin polymer. This correlation is observed for lignins of various degree of branching. Figures C.2c,d demonstrate that, in the limit of $\Delta \rightarrow 0$, the branched lignins have larger R_g than the unbranched at 300K, but the opposite trend is observed at 480K.

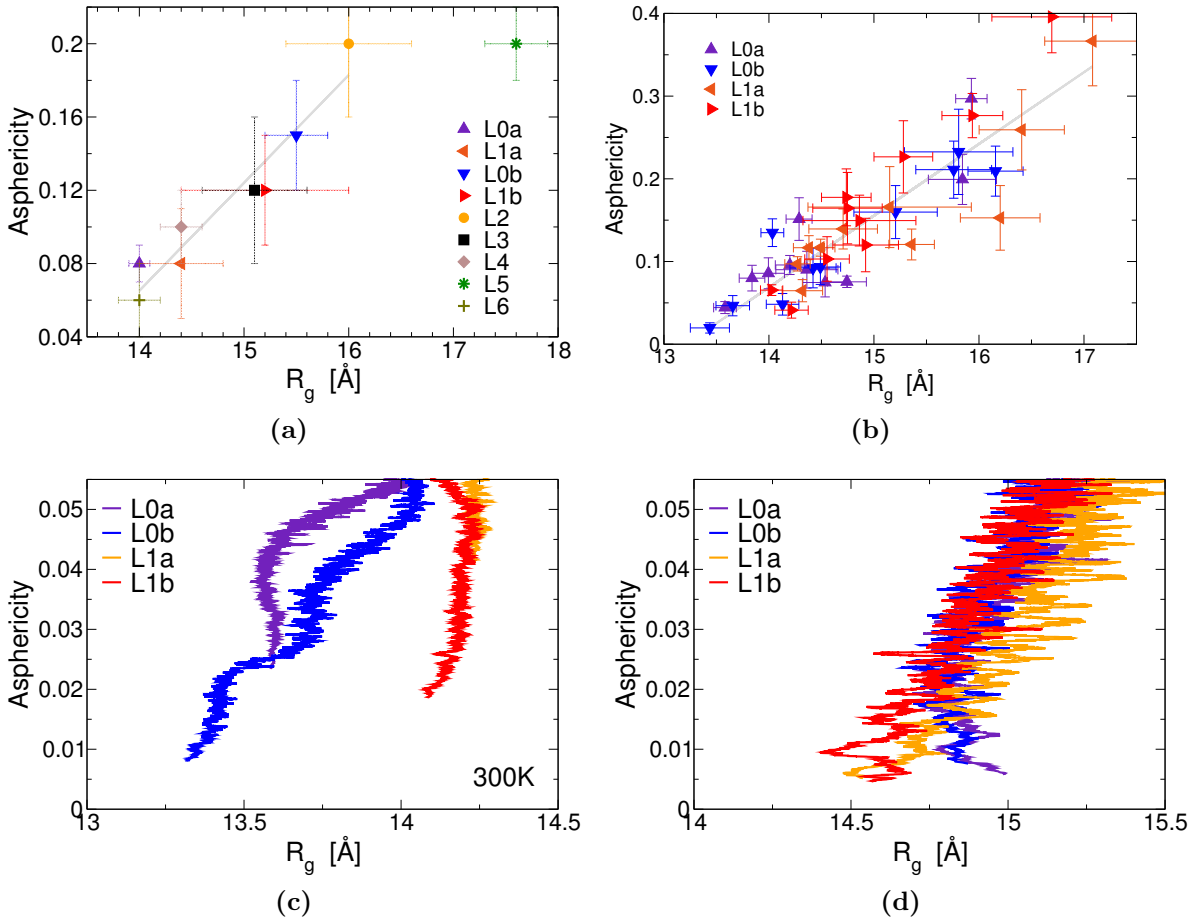


Figure C.4: (a) Average asphericity and radius of gyration for all nine lignins at $T = 300$K, values taken from Table 4.1; (b) Same as (a) but from the ensemble of lignins with zero and one branch points. (c) Scatter plot of asphericity against R_g from each frame of the ensemble of lignins with zero and one branch points. at $T = 300$K. Each point represents a running average over 100 frames and only data with $\Delta < 0.06$ are shown. (d) Same as (c), but at $T = 480$K.

C.7 Structure of Hydration Water

Structural properties of water close to the surface of the lignin are quantified by the proximal distribution function $g_{prox}(r)$, given by Equation 6. The density of the lignin hydration shell, ρ_{sh} was derived using Equation C.9:

$$\rho_{sh} = \frac{\eta_{sh}}{r_{max}} = \frac{\int_0^{r_{max}} g_{prox}(r)\ dr}{r_{max}}, \tag{C.9}$$

where η_{sh} is the integral of the proximal distribution function and r_{max} the position of the first minimum of $g_{prox}(r)$ that defines the outer boundary of the hydration shell. A similar definition can be obtained for the density of bulk water:

$$\rho_0 = \frac{\eta_0}{r_{max}} = \frac{\int_0^{r_{max}} g_{o-o}(r)\ dr}{r_{max}}, \tag{C.10}$$

with η_0 the integral of the standard oxygen-oxygen radial distribution function of bulk water. Taking $r_{max} = 4.9$Å in Equations C.9 and C.10 allows comparison of the hydration shell density with that of the bulk: $\rho_{sh} = \rho_0(\eta_{sh}/\eta_0)$. Figures C.5a-C.5d show the hydration shell density of lignin to be smaller than the bulk by 2%, 4%, 7% and 10% at T = 300K, 360K, 420K and 480K, respectively. This decrease in the hydration shell density is independent of geometric contributions that are also present if the hydration water is unperturbed from the bulk.

For the ensemble of nine lignins at $T = 300$K, with various degrees of branching, the average fraction of the SASA which is hydrophilic is $\langle\phi\rangle_{pol} = 0.43 \pm 0.01$, where a "hydrophilic" atom is crudely defined as having partial charge $|q| > 0.2e$. This is higher than the average fraction of the surface area of an isolated monomer that is hydrophilic: $\langle\phi\rangle_{mon} = 0.37 \pm 0.01$. $\langle\phi\rangle_{pol} > \langle\phi\rangle_{mon}$ indicates that hydrophilic hydroxyl moieties of lignin are preferentially exposed to the solvent in order to maximize favorable interactions with the water molecules. In a separate 80ns MD simulation of lignin $L0_a$ in vacuum the behavior was different due to burial of hydrophilic groups, with $\langle\phi^{vac}\rangle_{pol} = 0.36 \pm 0.01$ similar to $\langle\phi\rangle_{mon} = 0.37$.

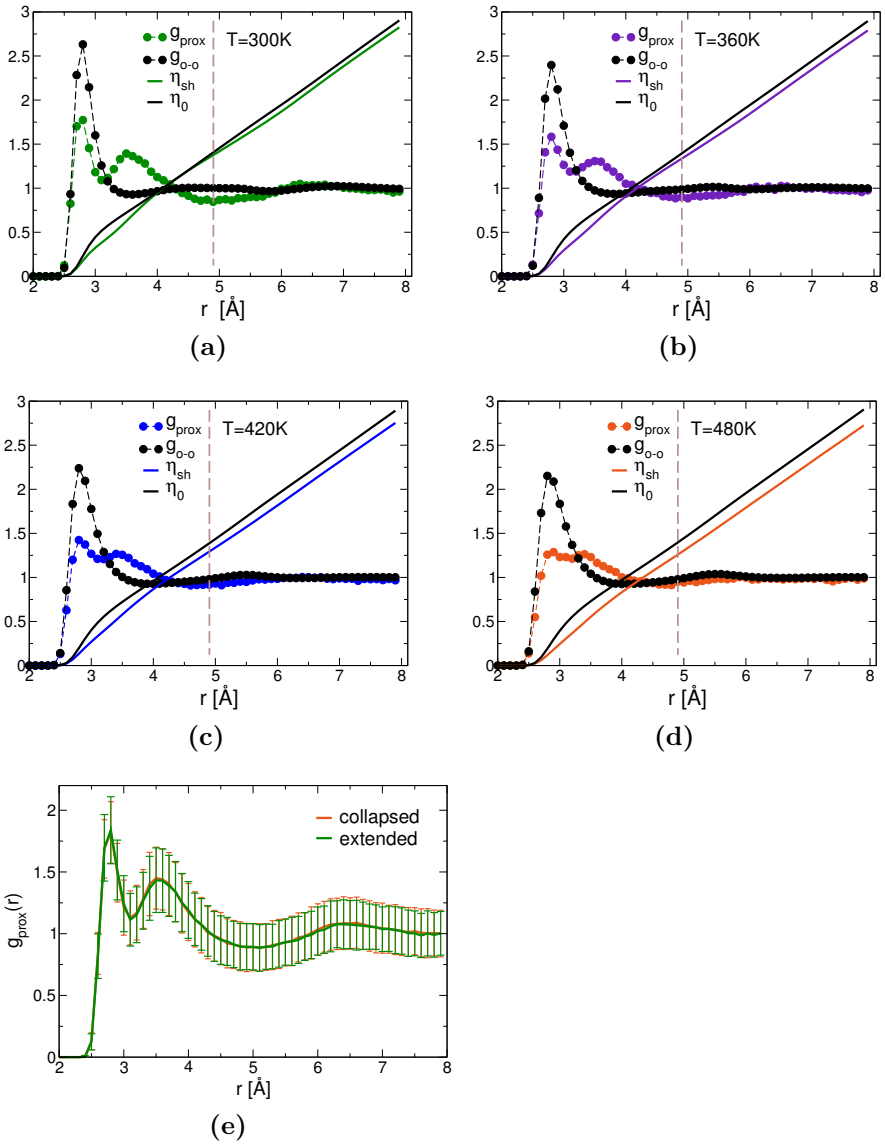


Figure C.5: Proximal distribution functions of the oxygen atoms of lignin hydration water (g_{prox}), radial distribution function of bulk water oxygen atoms (g_{o-o}) and the respective cumulative sums, η_{sh} and η_0. The dashed vertical line, $r = 4.9$Å, marks the outer boundary of the hydration shell of lignin. (e) Proximal distribution functions of water oxygen atoms at a distance r from the surface of the collapsed and extended lignins at $T = 300$K.

C.8 Enthalpy Change at 480K

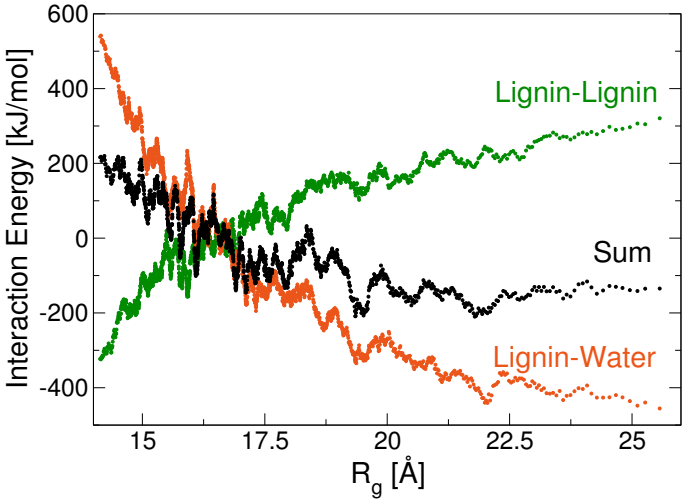


Figure C.6: Lignin-lignin and lignin-water interactions energies as a function of the lignin R_g at 480K. Data represent ensemble average of the unbranched and one-branch lignins.

C.9 Calculation of the Entropy of Water

Here, we provide the equations required to calculate the the entropy of water molecules from Eq. 12 of the main text. Full details on how these formulae are derived can be found in Refs Lin et al. (2003, 2010).

The density of states of the gas-like component is assumed to be that of hard spheres

$$g^g(\omega) = \frac{g_0}{1 + (g_0\omega/12fW)^2}, \tag{C.11}$$

where $g_0 = g(\omega = 0)$, W the number of water molecules, and f is the fluidity factor. f is determined by solving

$$2\delta^{-9/2}f^{15/2} - 6\delta^{-3}f^5 - \delta^{-3/2}f^{7/2} + 6\delta^{-3/2}f^{5/2} + 2f - 2 = 0, \tag{C.12}$$

and the normalized diffusivity δ is given by:

$$\delta = \frac{2g_0}{9N_{wat}} \left(\frac{\pi k_B T}{m}\right)^{1/2} \left(\frac{W}{V}\right)^{1/3} \left(\frac{6}{\pi}\right)^{2/3}, \tag{C.13}$$

where m is the mass of a water molecule and V the volume of the system.

The weighting functions of Eq. 12 are:

$$\lambda^s = \frac{\beta h\omega}{\exp(\beta h\omega) - 1} - \ln\left[1 - \exp(-\beta h\omega)\right], \tag{C.14}$$

where $\beta = 1/k_B T$ and h is Planck's constant; and

$$\lambda^g = \frac{S^{HS}}{3k_B}, \tag{C.15}$$

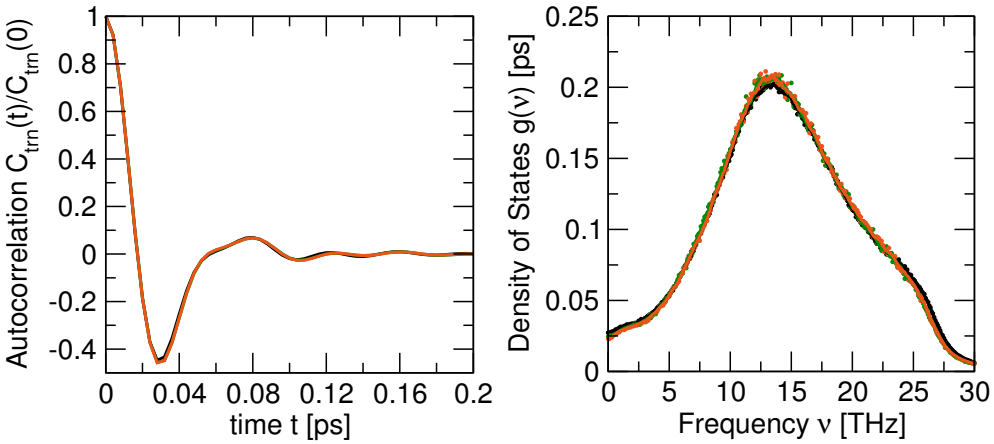


Figure C.7: (a) Rotational velocity autocorrelation functions and the respective (b) density of states of water.

where S^{HS} is the hard sphere entropy, given by

$$\frac{S^{HS}}{k_B} = \frac{5}{2} + \ln\left[\left(\frac{2\pi m k_B T}{h^2}\right)^{3/2} \frac{V}{fW} z(y)\right] + \frac{y(3y-4)}{(1-y)^2}, \tag{C.16}$$

where $y = f^{5/2}/\delta^{3/2}$ and

$$z(y) = \frac{1 + y + y^2 - y^3}{(1-y)^3}. \tag{C.17}$$

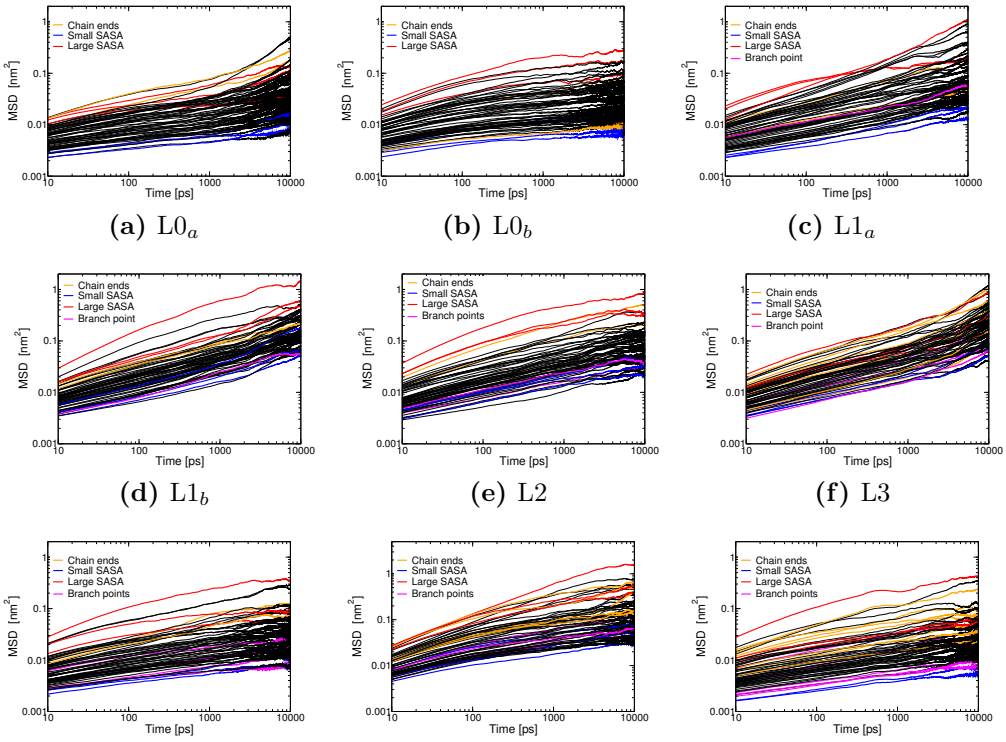
Substituting Eqs. C.11, C.14 and C.15 into Eq. 12, permits computation of the entropy of water molecules.

Figure C.7 shows the rotational velocity autocorrelation and $g(\omega)$ spectra, indicating almost no variation between the bulk and hydration water

C.10 MSD

Figure C.8 demonstrates the effect of solvent exposure and chain connectivity on monomer mean-square displacements of the nine lignin molecules of various branching at 300K.

Figure C.9 demonstrates the correlation between the monomer mobility and solvent exposure at T=360K and 420K.

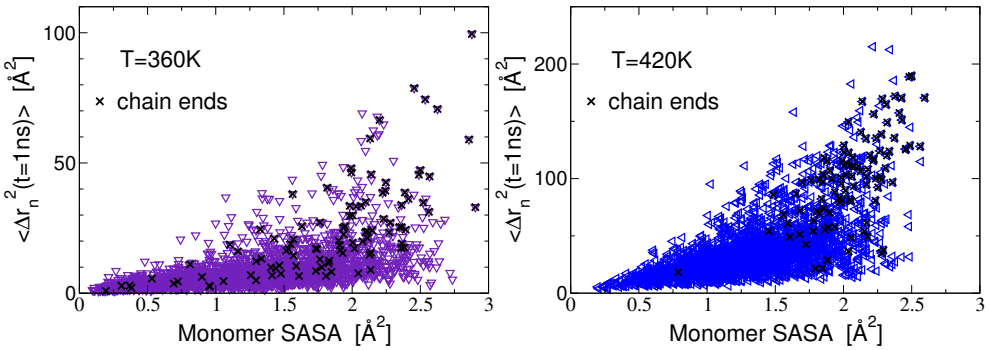


(a) L0$_a$ **(b)** L0$_b$ **(c)** L1$_a$

(d) L1$_b$ **(e)** L2 **(f)** L3

(g) L4 **(h)** L5 **(i)** L6

Figure C.8: Mean square displacement of lignins with zero (a) and (b), one (c) and (d), two (e), three (f), four (g), five (h), and six (i) branch points. Translation and rotation of the entire molecule have been removed. Highlighted monomers have the largest (red) and smallest (blue) SASA. MSD of monomers at chain ends are shown in orange and of branch points in magenta.



(a) 360K **(b)** 420K

Figure C.9: Monomer MSD at t = 1ns of the ensemble of polymers with zero and one branch points versus the monomer SASA at T=360K and 420K.

C.11 Chain Topology

Below are tables describing the linkages connecting the monomers of the nine simulated lignins. Each line contains the type of linkage (capital letters L and R refer to left- and right-handed linkages that contain a chiral center), followed by the numbers of the two monomers that are connected

Table C.2: Linkages connecting the monomers of lignin $L0a$.

type	monomer 1	monomer 2
55	1	2
$b5L$	2	3
$bO4L$	3	4
55	4	5
$aO4L$	5	6
$bO4R$	6	7
$bO4L$	7	8
$bO4R$	8	9
55	9	10
$bO4L$	10	11
55	11	12
$bO4R$	12	13
$b5L$	13	14
$b5R$	14	15
$bO4L$	15	16
$aO4R$	16	17
$bO4R$	17	18
55	18	19
$bO4L$	19	20
$aO4L$	20	21
55	21	22
$bO4R$	22	23
$bO4L$	23	24
55	24	25
$bO4R$	25	26
55	26	27
$bO4R$	27	28
$bO4L$	28	29
$bO4R$	29	30
55	30	31
$bO4L$	31	32
$bO4R$	32	33
55	33	34
$b5R$	34	35
$bO4L$	35	36
55	36	37
$bO4R$	37	38
$bO4L$	38	39
$bO4R$	39	40
$aO4R$	40	41
$bO4L$	41	42
55	42	43
$b5L$	43	44
$bO4R$	44	45
55	45	46
$bO4L$	46	47
$aO4R$	47	48
$bO4R$	48	49
$bO4L$	49	50
55	50	51
$aO4L$	51	52
55	52	53
$bO4R$	53	54
$b5R$	54	55
$bO4L$	55	56
55	56	57
$bO4R$	57	58
55	58	59
$bO4L$	59	60
55	60	61

Table C.3: Linkages connecting the monomers of lignin *L0b*.

type	monomer 1	monomer 2
55	1	2
$b O 4 L$	2	3
55	3	4
$b 5 L$	4	5
$a O 4 R$	5	6
$a O 4 L$	6	7
55	7	8
$b O 4 R$	8	9
55	9	10
$b O 4 L$	10	11
$b O 4 R$	11	12
55	12	13
$b O 4 L$	13	14
55	14	15
$b O 4 R$	15	16
55	16	17
$b O 4 L$	17	18
$b 5 R$	18	19
$b O 4 R$	19	20
55	20	21
$b O 4 L$	21	22
$b O 4 R$	22	23
55	23	24
$b O 4 L$	24	25
55	25	26
$b O 4 R$	26	27
$a O 4 R$	27	28
$b 5 L$	28	29
$b 5 R$	29	30
$b O 4 L$	30	31
$b O 4 R$	31	32
55	32	33
$b O 4 L$	33	34
55	34	35
$b O 4 R$	35	36
55	36	37
$b O 4 L$	37	38
$b O 4 R$	38	39
55	39	40
$a O 4 L$	40	41
$b O 4 L$	41	42
$b O 4 R$	42	43
$b 5 L$	43	44
$b O 4 L$	44	45
$b O 4 R$	45	46
$b 5 R$	46	47
$b O 4 L$	47	48
$b O 4 R$	48	49
55	49	50
$b O 4 L$	50	51
$b O 4 R$	51	52
$b O 4 L$	52	53
$b O 4 R$	53	54
55	54	55
$b O 4 L$	55	56
$a O 4 R$	56	57
55	57	58
$b O 4 R$	58	59
$a O 4 L$	59	60
55	60	61

Table C.4: Linkages connecting the monomers of lignin $L1a$.

type	monomer 1	monomer 2
55	1	2
bO4L	2	3
bO4R	3	4
bO4L	4	5
55	5	6
bO4R	6	7
bO4L	7	8
55	8	9
b5L	9	10
bO4R	10	11
55	11	12
bO4L	12	13
55	13	14
bO4R	14	15
aO4L	15	16
bO4L	16	17
bO4R	17	18
b5R	18	19
aO4R	19	20
55	20	21
bO4L	21	22
bO4R	22	23
bO4L	23	24
55	24	25
bO4R	13	26
b5L	26	27
bO4L	27	28
55	28	29
bO4R	29	30
55	30	31
aO4L	31	32
aO4R	32	33
55	33	34
b5R	34	35
bO4L	35	36
bO4R	36	37
bO4L	37	38
55	38	39
b5L	39	40
bO4R	40	41
55	41	42
bO4L	42	43
55	43	44
bO4R	44	45
bO4L	45	46
55	46	47
bO4R	47	48
bO4L	48	49
aO4L	49	50
bO4R	50	51
55	51	52
bO4L	52	53
55	53	54
b5R	54	55
aO4R	55	56
55	56	57
bO4R	57	58
bO4L	58	59
bO4R	59	60
55	60	61

Table C.5: Linkages connecting the monomers of lignin $L1b$.

type	monomer 1	monomer 2
55	1	2
$b O4L$	2	3
$b O4R$	3	4
55	4	5
$b O4L$	5	6
55	6	7
$b O4R$	7	8
$b O4L$	8	9
55	9	10
$b O4R$	10	11
55	11	12
$b O4L$	12	13
$b 5R$	13	14
$b O4R$	14	15
$b O4L$	15	16
$b O4R$	16	17
$a O4L$	17	18
$b O4L$	18	19
$b O4R$	19	20
55	20	21
$b O4L$	21	22
55	22	23
$a O4R$	23	24
55	24	25
$b O4R$	25	26
$b O4L$	26	27
$b 5L$	27	28
$b O4R$	28	29
55	29	30
$b O4L$	30	31
$a O4L$	31	32
55	32	33
$b O4R$	33	34
55	15	35
$b O4L$	35	36
55	36	37
$b O4R$	37	38
55	38	39
$b O4L$	39	40
55	40	41
$b 5R$	41	42
$b 5L$	42	43
$a O4L$	43	44
$b O4R$	44	45
$b O4L$	45	46
$b O4R$	46	47
55	47	48
$b O4L$	48	49
55	49	50
$a O4R$	50	51
$b O4R$	51	52
55	52	53
$a O4R$	53	54
$b O4L$	54	55
$b O4R$	55	56
$b 5R$	56	57
$b O4L$	57	58
55	58	59
$b 5L$	59	60
$b O4R$	60	61

Table C.6: Linkages connecting the monomers of lignin $L2$.

type	monomer 1	monomer 2
55	1	2
$b O 4 L$	2	3
$b O 4 R$	3	4
55	4	5
$b O 4 L$	5	6
$b 5 R$	6	7
$b O 4 R$	7	8
55	8	9
$b O 4 L$	9	10
55	10	11
$b O 4 R$	11	12
$b O 4 L$	12	13
$b O 4 R$	13	14
$b 5 L$	14	15
$a O 4 L$	15	16
55	16	17
$b O 4 R$	17	18
$b O 4 L$	18	19
$b O 4 R$	19	20
$a O 4 R$	20	21
$b O 4 L$	21	22
55	22	23
$b O 4 R$	23	24
55	24	25
$b O 4 L$	25	26
55	26	27
$a O 4 L$	27	28
$a O 4 R$	28	29
$b O 4 R$	29	30
$b O 4 L$	30	31
55	12	32
$b 5 R$	32	33
$b O 4 R$	33	34
55	34	35
$b O 4 L$	35	36
55	36	37
$b 5 L$	37	38
$b O 4 R$	38	39
$b O 4 L$	39	40
$b O 4 R$	40	41
$a O 4 L$	41	42
55	42	43
$b O 4 L$	43	44
$b O 4 R$	44	45
55	45	46
$b O 4 L$	46	47
$b O 4 R$	47	48
55	48	49
$b O 4 L$	49	50
$a O 4 R$	50	51
55	51	52
55	39	53
$b O 4 R$	53	54
$b O 4 L$	54	55
55	55	56
$b O 4 R$	56	57
55	57	58
$b 5 R$	58	59
$b 5 L$	59	60
$b O 4 L$	60	61

Table C.7: Linkages connecting the monomers of lignin $L3$.

type	monomer 1	monomer 2
55	1	2
$b O 4 L$	2	3
$b O 4 R$	3	4
55	4	5
$a O 4 L$	5	6
$b O 4 L$	6	7
$b O 4 R$	7	8
$a O 4 R$	8	9
$a O 4 L$	9	10
$b O 4 L$	10	11
55	11	12
$b O 4 R$	12	13
$b 5 L$	13	14
$b 5 R$	14	15
$b O 4 L$	15	16
55	16	17
$b O 4 R$	17	18
55	18	19
$b O 4 R$	19	20
$b O 4 L$	1	21
$b O 4 R$	21	22
55	22	23
$a O 4 R$	15	24
55	24	25
$b O 4 L$	25	26
$b O 4 R$	26	27
55	27	28
$b O 4 L$	28	29
$a O 4 L$	29	30
55	21	31
$b O 4 R$	31	32
55	32	33
$b O 4 L$	33	34
55	34	35
$b O 4 R$	35	36
$b 5 L$	36	37
$b O 4 L$	37	38
$b O 4 R$	38	39
55	39	40
$b 5 R$	40	41
$b O 4 L$	41	42
55	42	43
$b O 4 R$	43	44
$a O 4 R$	44	45
$b O 4 L$	45	46
55	46	47
$b O 4 R$	29	48
55	48	49
$b O 4 L$	49	50
$b O 4 R$	50	51
55	51	52
$b 5 L$	52	53
$b 5 R$	53	54
$b O 4 L$	54	55
55	55	56
$b O 4 R$	56	57
$b O 4 L$	57	58
55	58	59
$b O 4 R$	59	60
$b O 4 L$	60	61

Table C.8: Linkages connecting the monomers of lignin $L4$.

type	monomer 1	monomer 2
55	1	2
$b O4L$	2	3
$b O4R$	3	4
$b O4L$	4	5
$b O4R$	5	6
$b 5L$	6	7
$b O4L$	7	8
55	8	9
$b O4R$	9	10
55	10	11
$b 5R$	11	12
$b O4L$	12	13
$b O4R$	13	14
55	14	15
$b O4L$	15	16
55	16	17
$b 5L$	17	18
$b O4R$	18	19
55	19	20
$a O4L$	20	21
55	21	22
$b O4L$	22	23
55	23	24
$a O4R$	18	25
$a O4L$	25	26
$b O4R$	26	27
$b O4L$	27	28
55	28	29
$b O4R$	29	30
55	26	31
$a O4R$	31	32
$b O4L$	32	33
55	33	34
$b O4R$	34	35
$b O4L$	24	36
$b O4R$	36	37
55	37	38
$b O4L$	37	39
$b O4R$	39	40
$b 5R$	40	41
$a O4L$	41	42
55	42	43
$b O4L$	43	44
55	44	45
$b O4R$	45	46
$b O4L$	46	47
$b O4R$	47	48
$a O4R$	33	49
55	49	50
$b O4L$	50	51
$b O4R$	51	52
$b 5R$	52	53
$b 5L$	53	54
$b O4L$	54	55
55	55	56
$b O4R$	56	57
55	57	58
$b O4L$	58	59
55	59	60
$b O4R$	60	61

Table C.9: Linkages connecting the monomers of lignin $L5$.

type	monomer 1	monomer 2
55	1	2
bO4L	2	3
55	3	4
bO4R	4	5
bO4L	5	6
b5R	6	7
bO4R	7	8
55	8	9
bO4R	9	10
bO4L	10	11
aO4L	11	12
bO4R	12	13
55	13	14
bO4L	14	15
b5L	15	16
aO4R	16	17
55	17	18
bO4R	18	19
bO4L	19	20
aO4L	20	21
bO4R	21	22
55	22	23
aO4R	17	24
55	24	25
b5R	25	26
b5L	26	27
bO4L	27	28
bO4R	28	29
55	29	30
bO4L	30	31
bO4R	31	32
bO4L	32	33
b5R	33	34
bO4R	34	35
55	35	36
bO4L	36	37
bO4R	37	38
55	37	39
bO4L	39	40
bO4R	40	41
aO4L	27	42
55	42	43
bO4L	43	44
55	44	45
bO4R	45	46
b5L	46	47
bO4L	47	48
bO4R	48	49
bO4L	49	50
55	50	51
bO4R	51	52
bO4L	38	53
55	20	54
bO4L	54	55
55	55	56
55	19	57
bO4R	57	58
55	58	59
aO4R	59	60
55	60	61

Table C.10: Linkages connecting the monomers of lignin $L6$.

type	monomer 1	monomer 2
$b O4L$	1	2
$b O4R$	2	3
55	3	4
$b O4L$	4	5
$b O4R$	3	6
55	6	7
$b O4L$	7	8
55	8	9
$b5R$	9	10
$b5L$	10	11
$b5R$	11	12
$b O4R$	12	13
$b O4L$	13	14
$b O4R$	14	15
$b O4L$	15	16
$b O4R$	16	17
55	17	18
$a O4L$	11	19
$a O4R$	19	20
55	20	21
$a O4L$	21	22
55	22	23
$b O4L$	23	24
55	24	25
$b O4R$	25	26
$b5L$	26	27
$b O4L$	27	28
$b O4R$	28	29
$b O4L$	29	30
55	30	31
$b O4R$	31	32
55	32	33
$b O4L$	33	34
55	34	35
$b O4R$	35	36
55	36	37
$b O4L$	37	38
$b O4R$	38	39
$b O4L$	39	40
55	13	41
$b O4R$	41	42
55	42	43
$b O4L$	43	44
55	44	45
$a O4R$	45	46
$b O4R$	21	47
55	47	48
$b O4L$	48	49
$b O4R$	49	50
$a O4L$	50	51
55	51	52
$b O4L$	52	53
55	53	54
$b O4R$	54	55
$b O4L$	30	56
$a O4R$	56	57
55	57	58
$b5R$	58	59
$b5L$	53	60
$b O4R$	60	61

Appendix D

Additional performance results

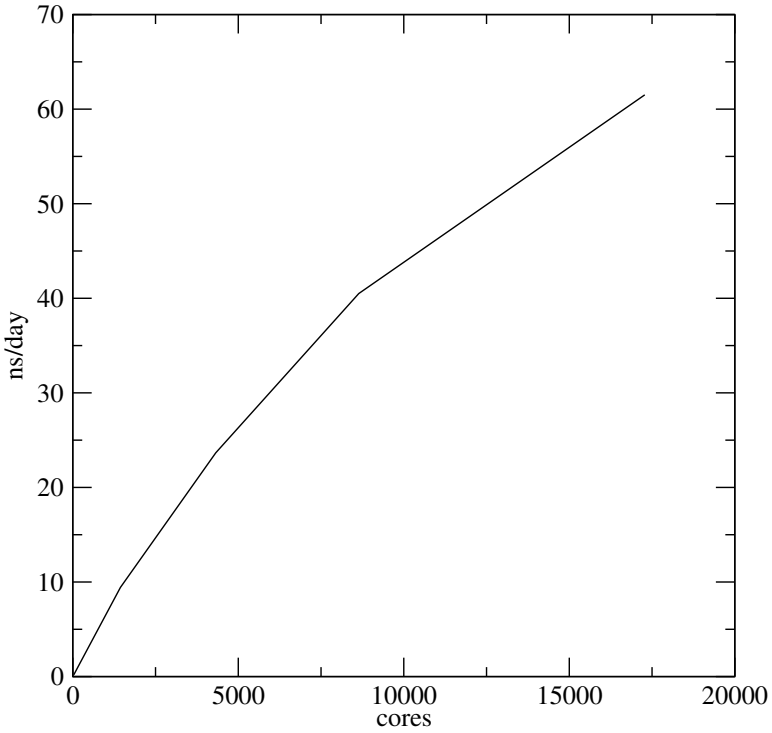


Figure D.1: Performance of 7.68 million atom ethanol-water system on Eos (Cray XC30) with a pre-release version of GROMACS 5.1. A timestep of 2fs, a cut-off of 1nm, neighbor-list update every 40 steps, and Fourier spacing of 0.125nm is used.

Vita

Roland Schulz was born in Duisburg, Germany, in 1980. In 2000, he graduated from high-school as valedictorian, winning a national prize for his academic achievements in physics. Roland pursued studies in Physics at the University of Braunscheig and Heidelberg. During this time, he cofounded Galilei Consult. In 2007, he joined the Center for Molecular Biophysics at the Oak Ridge National Laboratory and the Genome Science & Technology program at the University of Tennessee to obtain his doctoral degree in life sciences.